BIOINFORMATICS

第2版

The Machine Learning Approach

生物信息学

——机器学习方法

[法] 皮埃尔·巴尔迪（Pierre Baldi）

[丹麦] 索恩·布鲁纳克（Søren Brunak） 著

张东晖等 译

李衍达 朱宗涵等 审校

中信出版社

CITIC PUBLISHING HOUSE

当今图书市场上最好的生物信息学著作

本书作者不仅给我们展示了当今生物信息学大厦的缩影和构筑大厦的工具，更重要的是作者带领我们经历了如何构筑这个大厦，如何搭建"脚手架"的过程，这无论对于修补这座大厦还是构建一座新的大厦都是非常重要的。

"这是一本生物信息学家不可或缺的藏书。"

——特里·卡斯特朗 《自然生物技术》杂志

"作者将十分广泛和深入的素材组织得如此清晰易读且令人信服，这确实是一本编排精巧、内容丰富的好书。"

——马克·扶特利 著名生物学家

"仅靠这一本书或许很难掌握生物信息学的全部内容，但如果你想理解生物信息学，此书是不可不读的。"

"本书是现今图书市场上最好的一本生物信息学著作。"

——《亚马逊书评》





ISBN 7-80073-708-X/Q·1

定价：45.00元

BIOINFORMATICS

The Machine Learning Approach

生物信息学

——机器学习方法

[法] 皮埃尔·巴尔迪

[丹麦] 索恩·布鲁纳克　著

张东晖 黄　颖 蔡　军 孙应飞 夏慧煜 胡驰峰 计宏凯 朱宗涵　译

李衍达 朱宗涵 张东晖　审校

本书翻译工作得到《国家重点基础研究发展规划》课题（编号：2001CB51030）的支持

中信出版社

CITIC PUBLISHING HOUSE

图书在版编目（CIP）数据

生物信息学 / [法] 巴尔迪等著；张东晖等译；李衍达等审校. —北京：中信出版社，2003.5

书名原名：Bioinformatics: The Machine Learning Approach

ISBN 7-80073-708-X

Ⅰ. 生…　Ⅱ. ①巴…　②张…　③李…　Ⅲ. 生物信息论　Ⅳ. Q811.4

中国版本图书馆CIP数据核字（2003）第032756号



The MIT Press

http://mitpress.mit.edu

生物信息学——机器学习方法

SHENGWU XINXIXUE——JIQI XUEXI FANGFA

著　　者：[法] 皮埃尔·巴尔迪　[丹麦] 索恩·布鲁纳克

译　　者：张东晖　黄　颖　蔡　军　孙应飞　夏慧煜　胡驰峰　计宏凯　朱宗涵

审 校 者：李衍达　朱宗涵　张东晖

责任编辑：陈蕴真

出 版 者：中信出版社（北京市朝阳区东外大街亮马河南路14号塔园外交办公大楼　邮编　100600）

经 销 者：中信联合发行有限公司

承 印 者：北京忠信诚胶印厂

开　　本：787mm × 1092mm　1/16　　印　　张：26.75　　字　　数：342千字

版　　次：2003年7月第1版　　印　　次：2003年7月第1次印刷

京权图字：01-2002-0211

书　　号：ISBN 7-80073-708-X/Q · 1

定　　价：45.00元

译者序

2002年夏天，中信出版社交给我一本英文原著，是由皮埃尔·巴尔迪（Pierre Baldi）和索恩·布鲁纳克（Søren Brunak）两位教授编写的《生物信息学——机器学习方法》（第2版），MIT出版社于2001年出版。出版社的编辑同志告诉我，鉴于本书的学术价值及其在生物信息学领域的重要性，出版社已购买了本书的中文版权，并准备作为社里的重点图书尽快在国内翻译出版。由于本书作者在生命科学、数学以及计算机科学等多个领域都有相当的造诣，加之本书同时涉及了生物信息学的理论基础和最前沿的实际应用，出版社走访了几位专家译者，他们都不愿意承担这一艰巨的翻译工作。我用了整整一个星期的时间，认真阅读了这本书的前言、目录和一些重要章节，深感本书分量之重。在此之前，我也曾经读过几本国内出版的生物信息学著作或译著，其中大部分是有关基因和蛋白质序列分析软件、算法以及相关网络资源的工具书，而真正涉及生物信息学基本理论和最前沿应用的著作还很少。我们在实际工作中经常会利用国外的一些生物信息学的数据库和软件，分析基因或蛋白质的序列和结构，但对于这些数据库和软件背后的理论、模型和算法却所知甚少。随着国内生物信息学和生命科学等相关领域研究工作的不断深入和发展，我们的研究方向已经从积累数据和追踪国外最近进展逐步转向前沿的基础研究和新的应用开发，而这些前沿领域的研究和开发要求我们了解和掌握生物信息学的主要理论、模型和算法。为了适应这些新的研究方向，越来越多的本科和研究生专业已经或将要开设生物信息学课程。因此，国内的生物信息学领域迫切需要一本足够深入的经典教材或参考书，而本书正好可以满足这一迫切需求。正如国外专家对本书的评论中所说的："仅靠这一本书或许很难掌握生物信息学的全部内容，但如果你想理解生物信息学，此书是不可不读的。"为此，我决定接受翻译此书这一艰巨的任务。

本书的内容涉及生命科学、数学、信息科学等诸多领域的最新进展，我深知仅靠

我个人很难在短期内完成全书的翻译工作，必须邀请相关领域的专家组成翻译小组，合作完成全书的翻译和审校工作。于是，我找到了我国著名的信息科学专家，清华大学信息学院院长、生物信息学研究中心主任李衍达院士，他欣然同意主持本书的翻译工作。我们还邀请到微软（中国）公司的资深软件设计工程师张东晖先生，以及清华大学信息学院的黄颖、蔡军等多位博士，共同组成翻译小组，几易其稿，又请了多位专家参与审校，最终完成了本书的中文译稿。整个过程的艰辛难以用语言表达。为此，要感谢李衍达院士的全力支持和翻译小组全体同仁付出的宝贵精力和时间，也要感谢中信出版社青年编辑陈缢真同志的真诚合作。本书的翻译还得到了“国家重点基础研究发展规划”课题（编号：2001CB51030）的支持，北京市卫生局干部培训中心为翻译小组提供了良好的工作条件，在此一并致谢。

本书的作者是国际著名的生物信息学专家。其中皮埃尔·巴尔迪博士是美国加州大学医学院信息和计算机科学系教授、生物化学系教授，基因组学和生物信息学研究所所长。索恩·布鲁纳克博士是丹麦理工大学生物系教授，生物序列分析中心主任。他们在生物信息学领域发表了大量的论文和著作，涉及到许多生物信息学理论、模型和算法的前沿应用和探索。本书是他们多年研究和教学工作的积累，本书的早期版本曾作为几个国际重要的生物信息学研讨班的讲义。他们在本书中详细介绍了机器学习方法的理论基础——贝叶斯概率体系，并在此基础上着重讨论了神经网络、隐马氏模型、贝叶斯网络、概率图模型以及随机文法等不同方法在序列比对、基因建模与基因发现、系统进化树等生物信息学问题中的应用。书中还专辟一章介绍了DNA微阵列和基因表达，以及相关数据的分析方法。此外，本书还分类列举了大量相关网络资源的详尽网址，以及近600条参考文献和5个包含详尽数学推导的附录，这些参考资料无疑会给生物信息学的研究和教学工作者提供非常实际的帮助。

在翻译和审校过程中，我们发现本书有几个值得关注的特点。首先，本书试图利用贝叶斯概率理论的统一框架为机器学习方法在生物信息学领域的应用建立一套完备的理论基础。书中使用大量篇幅介绍了如何在概率理论的统一框架内理解神经网络、隐马氏模型、概率图模型以及随机文法等机器学习方法，并详尽地介绍了各种建模和学习算法。作者对理论完备性的追求无疑给读者提供了很好的背景知识和扎实的理论基础。第二，本书不仅介绍了机器学习的理论和算法，还介绍了大量的实际应用。最宝贵的是书中包含了作者在应用各种机器学习方法解决实际问题中所做的深入观察、富有创造力的假设、精细的建模、真实的实验结果和透彻分析，以及不断修正假设和模型的整个探索过程。与许多流行的教科书不同，作者不仅给我们展示了当今生物信息学大厦的缩影和构筑大厦的工具，更重要的是作者带领我们经历了如何构筑这个大厦的过程，如何搭建“脚手架”的经验无论对于修补这座大厦还是构建一座新的大厦都是非常重要的。第三，

正如许多专家的评论所指出的，本书在介绍相关理论和应用的同时，还提出了生物信息学前沿领域的许多重要问题。读者不仅可以了解生物信息学的前沿领域，还可以追随原作者探索这些问题的轨迹，开始自己对前沿问题的开创性研究。

本书主要针对两类读者：一类是生物学、生物化学和医学等领域的研究人员，他们可以通过本书了解更多数据处理和机器学习的有关算法；另一类是物理、数学、统计学和计算机科学等领域的学者，他们也可以通过本书了解机器学习方法在生命科学，特别是在生物信息学领域中的更多应用。本书也可以作为相关领域的大学本科和研究生教材或参考读物。

最后，我想指出尽管本书在建立生物信息学的理论基础方面可谓是一次成功的尝试，但生物信息学作为一门新兴的跨学科的科学还处在起步阶段。在本书的翻译过程中，我们深刻地体会到来自不同学科研究人员之间的密切合作和相互理解是多么重要。虽然本书包含许多数学公式和推导，但这并不意味着生物信息学排斥那些不熟悉数学公式的生物学和医学专家，如果失去了生物学和医学专家的合作与理解，生物信息学将失去继续发展的动力和应用的基础。为此，我们真诚地希望本书能够赢得来自生物学与医学领域专家的更多理解与关注。

虽然我们尽了很大努力，以确保翻译的质量和译文的准确，但是错误之处在所难免，希望广大读者批评指正。

朱宗涵

2003年3月20日

中文版序

我们很高兴看到自己的著作《生物信息学——机器学习方法》的第2版翻译成中文。中国在人类基因组计划中做出了重要贡献，水稻基因组的测序也给世界留下了深刻的印象，包括猪基因组测序在内的一系列国际交流合作都证明中国在基因组研究上达到了很高的水平。本书的出版更从另一个方面证明了这一点。作为一个拥有悠久历史的国家，中国在当今基因组研究的浪潮中具有自己独特的优势和发展机遇。现在，中国正迅速将先进的计算科学用于高通量的基因组和后基因组技术，并与那些传统的生物技术相结合。我们很荣幸此时能够为生物信息学研究思想的全球交流以及中国下一代计算生物学家的培养，尽自己的一份绵薄之力。现在中国的研究人员与世界具有密切的国际合作，我们希望本书的出版能够增进中国国内以及中国与其他国家之间的生物信息学理论和实验研究的协调合作。

皮埃尔·巴尔迪
索恩·布鲁纳克
2003年5月

前　言

本书第1版出乎意料的成功曾使我们深感欣慰。然而，由于生物信息学持续迅速发展，本书需要一个新的版本。在过去的3年里，随着果蝇基因组测序和人类基因组工程第一个草图的完成，全基因组测序研究蓬勃发展。除此之外，其他一些高通量/组合实验技术，如DNA微阵列（基因芯片）、质谱技术等，都取得了重大进展。这些高通量的实验技术能够快速产生10^{12}字节的实验数据，拥有传统生物学方法无法比拟的优势。这一切导致了今天对计算机、统计学和机器学习技术日益强烈的需求。

后基因组时代的生物信息学

在过去5到10年中，计算机在生命科学和医学的各个领域中发挥着前所未有的重要作用。计算机分析应用的第一个高潮主要出现在序列分析中，这个方面至今有许多非常重要的问题尚未解决；在目前以及未来的一段时期内，我们尤其需要关注那些极为多样化的数据的复杂集成关系。这些新的数据类型来源于能够在细胞、器官、生物个体甚至生物群体等不同层次获取数据的各种实验技术。

新的高效实验技术，主要是DNA测序技术，是以下转变的主要动力：新技术导致描述DNA、RNA和蛋白质的线性序列数据呈几何级数增长。其他新的产生数据的技术则是传统试验方法的高度并行版本。用DNA微阵列进行基因组范围的基因表达测定，基本上如同进行上万个RNA印迹实验（northern blots），这使得在实验设计、数据处理和结果解释等方面的计算机支持成为基本要求。而这一系列的发展极大地扩展了生物信息学的研究领域。

随着基因组和其他测序项目的不断进展，研究的重点正逐步从积累数据转移到如

何解释这些数据。在未来，生物学的新发现将极大地依赖于我们在多个维度和不同尺度下对多样化的数据进行组合和关联的分析能力，而不再仅依赖于对传统领域的继续关注。序列数据将与结构和功能数据、基因表达数据、生化反应通路数据、表现型和临床数据等一系列数据相互集成。在数据量呈几何级数增长的情况下，生物信息学的基础研究将致力于解决生命科学中与系统和集成相关的问题。

如此大量的数据，在生物信息的存储、获取、联网、处理、浏览以及可视化等方面，都对理论、算法和软件的发展提出了迫切的需求。而计算机科学也从生命系统中获得启示，产生了许多新概念，包括：遗传算法、人工神经网络、计算机病毒和人造免疫系统、DNA计算、人工生命以及VLSI-DNA混合基因芯片，等等。这样的学科交叉丰富了各个相关领域，这将在未来的几十年中得到进一步发展。事实上，基于“碳”的生物体信息处理和基于“硅”的电子化信息处理之间的界限，无论是在概念上，还是在实际中，都已开始逐渐淡化。[29]

用于序列分类、弱相似性探测、区分DNA序列中的编码区和非编码区、分子结构预测、转录后修饰和功能的预测，以及重构进化史的计算工具已经成为研究的基本组成部分。这些研究是我们理解生命和进化，以及发现新药物和新疗法的基础。生物信息学已成为在生命科学和计算机科学的前沿涌现出的一门具有战略意义的新学科，它将通过各种途径影响医学、生物技术以及社会的许多领域。

庞大的生物信息数据库对数据挖掘技术提出了许多颇具挑战性的问题，也提供了广阔的机遇，这些都需要研究人员提出新的思想和方法。在这方面，传统的计算机科学算法曾有用武之地，但面对许多最具重要意义的序列分析问题，它们越来越显示出不足。这一方面是由于进化不断修补基因，导致生物系统内在的复杂性；另一方面则由于我们尚缺乏一套在分子水平上理解生命组织的完整理论。而机器学习方法［例如神经网络、隐马氏模型、支持向量机、置信网络（belief network）］正适合这类数据量大、含有噪声模式并且缺乏统一理论的领域。机器学习方法的基本思想是通过推理、模型匹配或样本学习，从数据中自动学习理论。因此，机器学习方法是传统方法的重要发展。本书旨在从机器学习的角度对生物信息学进行广泛全面的介绍。

机器学习方法的计算量极大，因此在很大程度上得益于不断提高的计算机处理速度。值得注意的是，自20世纪80年代晚期以来，计算机的处理速度和序列数据量几乎以相同的速度增长，即大约每16个月增长1倍。而最近，随着人类基因组工程第一个草图的完成，以及诸如DNA微阵列等高效实验技术的出现，生物信息数据以更快的速度增长，每6~8个月就增长1倍，从而给生物信息学带来了更大的压力。在初学者看来，机器学习方法好像是一些彼此无关的技术的集合，其实并非如此。在理论方面，一个关于所有机器学习方法的统一的理论体系在20世纪80年代晚期已经产生，这就是用于建

模和推断的贝叶斯概率体系。实际上，在我们看来，机器学习方法与贝叶斯统计建模和推断之间，除了前者更强调计算机技术和大规模数据处理之外，几乎没有差别。正是由于数据、计算机和概率理论体系三者的交汇，才使得机器学习方法在生物信息学和其他领域获得了强劲的发展动力，并且不断扩展。客观地讲，生物信息学和机器学习方法已经开始在生物学和医学领域产生显著的影响。

即便您对数学的严格性缺乏敏感，生物数据的概率建模仍然具有重大意义。这一方面由于生物测量经常包含难以去除的噪声，例如目前的DNA微阵列或质谱数据等。另一方面，序列数据因其离散性质及重复测序的成本较低，并不受噪声约束。因此，测量噪声并非采用概率建模的惟一原因。对生物数据进行概率建模的真正需要来源于生命系统的复杂性和多样性，这一切来自于漫长的进化进程中生物体在复杂环境下历经的进化修补。这样的生命系统必然呈现很高的维度（dimensionality）。即使在能够同时测量数以千计的基因表达的微阵列试验中，我们也仅仅观察到相关变量的一个很小的子集，而其他绝大部分变量则仍然处于隐藏状态，我们必须依赖概率建模来确定它们。直接应用系统化的概率体系能够加速发现变量的过程，避免重复历史上序列分析所走过的弯路。概率模型作为正确的理论体系正是从序列分析这个过去几十年中充满荆棘的领域中逐步发展而来的。

机器学习技术经常受到的批评是，它们都是“黑箱”方法：我们总是无法确定一个复杂的神经网络或隐马氏模型是如何达到特定解的。我们已经尝试在全面的概率体系中以及从实践的角度解决这一问题。然而，我们需要看到，许多当代分子生物学的技术是完全基于经验的。例如聚合酶链式反应（PCR），就其实用性和灵敏度而言，在某种程度上仍然是一项黑箱技术，实验中许多参数调整仍然是通过尝试得到的。另一个例子是关于序列在胶体矩阵中的运动方式和机动性，这里人们更关心实际成功和可用性，而很少关心对其中物理现象细节的理解。同样，对大部分药物来说，其药理作用的分子基础目前在很大程度上尚属未知。理论最终需要实践检验。至此，我们已经简要地概述了机器学习方法的功能及其优势。

读者及预备知识

本书面向不同背景的学生以及高级研究人员。我们试图为具备较强数学、统计学和计算机科学背景的读者提供生物学基本概念和问题的阐述。同样地，本书内容的选择也考虑到生物学家和生物化学家的需要，他们的生物学知识超出本书的内容，但在理解生物数据处理的一些新算法方面需要更多的帮助。为了使读者能够实现本书中所介绍的算法或将算法应用于特定的问题，本书在提供相当深入的内容的同时试图保持足够的简练性。然而，我们并未涉及有关大型数据库和测序项目的管理，以及原始荧光数据处理

等方面的内容。本书对预备知识的要求包括大学本科水平的微积分、代数和离散概率理论等。任何关于DNA、RNA和蛋白质方面的知识都是有帮助的，但不是必需的。

内容提要

我们试图使本书成为一本全面深入且简练易读的介绍性著作。书中包括主要概念的定义和主要定理，它们至少是概略性。更多的技术细节可以在附录和参考文献中找到。本书的大部分内容基于我们在过去几年中发表的论文，以及在ISMB（Intelligent Systems for Molecular Biology）大会等会议上的讲义，在丹麦理工大学（Technical University of Denmark）、加州大学欧文分校（University California Irvine）以及在NIPS（Neural Information Processing Systems）会议期间组织的讨论班讲授的有关课程。尤其是作为本书核心的广义贝叶斯概率理论体系，曾在1994年之后的几届ISMB大会上讲解过。

本书主要介绍生物信息学领域的相关方法，而不是阐述这个迅猛发展的学科的历史。当我们引用相关文献的细节时，只将注意力集中于介绍相关技术以及一些通用的一般性思考方法。同时，我们试图用一些实验结果来说明每种方法，其中一些结果直接来源于我们自己的工作。

第1章　本章介绍分子生物学中的序列数据和序列分析。其中包括基因组和蛋白质组的概述，进化所创造的DNA和蛋白质数据，这些数据正逐步进入的这个领域的公共数据库。本章还包括基因组及其规模的概述，以及一些在其他教科书中很难找到的相关资料。

第2章　本章旨在建立整个机器学习技术的理论基础，并且介绍了存在不确定性的情况下如何进行推理，因此本章是有关理论的最重要的一章。本章阐述了序列问题的一般性思想方法：用于归纳和推理的贝叶斯统计理论体系。这一体系的主要观点是，概率理论语言是适合于处理机器学习及所有建模问题的语言。所有的模型必须是基于概率的。在科学地描述模型及其与数据间的关系时，概率理论是惟一需要的工具，这一点在本书的书名中已有所体现。本章简要涵盖了一些经典论题，如：先验分布、似然度、贝叶斯定理、参数估计和模型比较。在贝叶斯体系中，人们最关心的是与数据、隐变量以及模型参数等相关的高维空间中的概率分布。为了处理和逼近这些概率分布，需要尽可能地利用独立性假设以便进行简单的因子分解。图模型正是基于这一思想，模型中变量之间的依赖关系对应于图的连通性。一些易于求解的常用模型往往对应于相对稀疏的图。本章对图模型和其他一

些处理高维分布的技巧只做了简略介绍，更深入的内容参阅附录C。应用概率理论和（稀疏的）图模型必然成为各种方法的两个真正核心的思想。

第3章 本章用一些例子进一步说明广义贝叶斯概率体系，为以后的学习做准备。这里介绍了几个经典例子的处理细节，随后的几章中将用到它们。熟悉这些例子的读者在快速浏览本书时可跳过本章。本章中所有的例子都基于投掷一个或多个骰子从而生成序列的思想。骰子模型只是一个极为简单的模型，然而本书的主要部分，从第7章到第12章，都可以视为这个模型的不同推广。统计力学也被视为骰子模型在贝叶斯概率体系中的一个精彩应用。此外，统计力学在机器学习的许多方面为我们提供了深刻的启示。尤其在第4章，统计力学被应用于一系列算法中，如蒙特卡罗方法（Monte Carlo）和期望最大（expectation maximization，EM）等算法。

第4章 本章简要介绍了许多应用于贝叶斯推断、机器学习和序列分析的基本算法，这些算法大多用于计算期望值和优化代价函数（cost function）。这些算法包括各种形式的动态规划、梯度下降法和EM算法，以及一些随机算法，如马尔可夫链—蒙特卡罗算法（Markov Chain Monte Carlo，MCMC）。MCMC算法的一些著名应用，如吉布斯采样（Gibbs sampling）、Metropolis算法、模拟退火算法（simulated annealing）等，在本章中都有所涉及。在初次阅读时可以跳过本章，尤其是熟悉算法或者对算法的实现细节不感兴趣的读者。

第5章 第5~9章和第12章构成了本书的核心部分。第5章主要介绍神经网络的理论。其中包括基本概念的定义，反向传播学习算法的简要推导，以及神经网络作为广义函数逼近器的简单证明。更重要的是介绍了如何从第2章建立的一般概率体系出发，更好地理解神经网络这个经常被视为与概率理论不相关的方法。接下来，这种思想将用来指导神经网络结构设计以及机器学习中代价函数的选择。

第6章 本章列举了一些精心选择的应用神经网络技术解决序列分析问题的例子。我们并不想涵盖迄今为止的数百个应用例子，而只选择了一些由于方法论上的进展而显著改善了应用效果的范例。我们尤其关注那些序列分析中机器学习过程优化的问题，以及如何组合网络以构成更加全面有效的算法。本章中具体分析的方法包括：蛋白质的二级结构、信号肽内含子剪接位点和基因发现。

第7章 第7~8章是关于隐马氏模型（HMM），其内容安排与第5~6章相似。其中第7章包括对隐马氏模型的详尽介绍，相关的动态规划算法（前/后向算法和

Viterbi算法）和学习算法（EM算法、梯度下降法等）。生物序列的隐马氏模型可以理解为包含插入和删除操作的骰子模型的推广。

第8章 本章包括精心选择的隐马氏模型在蛋白质和DNA/RNA序列问题上的应用范例。这些例子示范了隐马氏模型的主要应用，即蛋白质家族建模、生成大规模多重序列比对、序列分类，以及在大型数据库中搜索完整或破碎的序列片断。对于DNA序列问题，我们介绍了隐马氏模型如何用于基因发现（启动子、外显子和内含子）和基因结构分析（gene-parsing）等任务。

第9章 尽管隐马氏模型非常有效，但它仍然存在一些局限性。第9~11章的内容可以看做隐马氏模型在不同方向上的扩展。其中第9章系统应用概率图模型的理论作为统一的概念，并从中导出几类新的模型，例如：隐马氏模型和神经网络相结合的混合模型，能够利用序列空间特征而不仅仅是时间特征的双向马尔可夫模型。本章还包括基因发现、DNA对称性分析和蛋白质二级结构的预测等应用。

第10章 本章介绍了系统进化树（phylogenetic tree）并将其纳入第2章建立的概率理论体系，由此导出进化的概率模型。本章讨论的模型以及本书的其他模型均可视为第3章中简单骰子模型的推广。我们特别指出：在了解这些方法所近似的内在概率模型的情况下，那些经常从非概率意义的角度阐述的系统进化树重构方法［如吝啬法（parsimony method）］，实际上只是广义概率体系的一个特例。

第11章 包括正则文法（formal grammar）和乔姆斯基层次（Chomsky hierarchy）。随机文法（stochastic grammar）作为隐马氏模型和简单骰子模型的推广，为生物序列提供了一类新的模型。其中随机正则文法（stochastic regular grammar）本质上等价于隐马氏模型。而上下文无关随机文法（stochastic context-free grammar）则有更强的表达能力，它大致对应于能够产生1对字符（而不只是1个字符）的骰子模型。本章简要回顾了随机文法的应用，尤其在RNA建模方面的应用。

第12章 本章主要集中于DNA微阵列的基因表达数据分析，并再一次推广了骰子模型。我们介绍了如何系统应用贝叶斯概率体系对微阵列数据进行分析。我们特别考虑了基因在不同条件下表达水平是否发生变化和基因聚类问题。本章还简要讨论了基因调控区的分析和基因调控网络的推导问题。

第13章 本章包括当前因特网上有关数据库资源和其他公共资源的概述，以及一个包含许多重要网站的网址目录和链接。由于这些资源变化很快，因此我们主要介绍一些定期更新信息的网站。当然，本章也给出了一些包含

其他相关网站链接的定期更新的网页。

本书的附录包含几节技术性较强的讨论，它们是深入理解本书内容的重要参考。

附录A　包括误差带（error bar）、充分统计量以及指数型分布族等统计学概念。

附录B　主要包括信息论以及熵、互信息（mutual information）、相对熵（relative entropy）等一些基本概念。

附录C　简要概述图模型、独立性和马尔可夫性，其中既包括无向图模型（随机马尔可夫域），也包括有向图模型（贝叶斯网络）。

附录D　关于隐马氏模型的一些技术问题，包括数域缩放（scaling）、环状构架（loop architecture）和可弯曲性（bendability）。

附录E　简要概述了两类相关且日趋重要的机器学习模型：高斯过程和支持向量机。

本书还附有许多练习题，从一些简单的证明到一些定理的扩展方法都有。

为了阐述方便起见，我们有时隐含了一些关于正定性或可微性的标准假设，但读者可以从上下文中清楚地知道这些假设成立。

第2版增加和删去的内容

在书中不同部分，我们增加了一些新的内容或者从一个新的角度对于原有内容进行阐述。例如，第3章中关于最大熵的讨论和关于波耳兹曼-吉布斯（Boltzmann-Gibbs）分布的推导；第8章中将隐马氏模型应用于序列片断、启动子、亲水性分布图（hydropathy profile）、可弯曲性分布图（bendability profile）分析；第10章中从概率论的角度分析吝啬法和高阶进化模型；第12章中关于芯片数据的基因差异表达的贝叶斯分析。另外，我们还给出了从自由能的角度看待EM算法。这种提法不为人熟知，根据我们得到的材料，这种方法最早是由尼尔（Neal）和欣顿（Hinton）在他们未发表的技术报告中提出的。

在本书第2版出版的过程中，我们从许多同事、学生和读者那里得到了大力的帮助和有益的反馈。书中许多地方都有不同程度的修正和更新，以便反映全基因组测序和其他高通量技术所引发的科学发现的迅速发展。此外，我们还在第2版中做了如下一些重大改变：

- 第1章中新增了介绍人类基因组序列的部分。
- 第1章中增加了关于蛋白质功能和可变剪接的内容。
- 第6章中列出了神经网络的一些新应用。
- 完全改写了第9章，其主要内容改为图模型的系统阐述及其在生物信息学中的应

用。本章还特别包含了有关基因发现，递归神经网络用于蛋白质二级结构预测的新内容。

- 增加了新的一章（第12章），专门讨论DNA微阵列数据和基因表达。
- 增加了一节新的附录（附录E），讨论支持向量机和高斯过程。

本书的材料组织和一些问题讨论反映了作者的个人偏好。由于篇幅所限，省略了一些相关问题的讨论。关于贝叶斯推断和贝叶斯网络的分析，在理论水平上尚待提高。如果从统一的角度出发和更有利于对问题进行抽象，本书的大部分内容实际上完全可以只用贝叶斯网络的思想加以组织写作。我们关于系统进化树、DNA微阵列和基因聚类的生物学讨论，还可以进一步扩充。无论如何，在相关问题具有合适的补充材料时，我们列出了丰富的参考文献。

词汇和表示法

诸如“生物信息学”（bioinformatics）、“计算生物学”（computational biology）、“计算分子生物学”（Computational molecular biology）以及“生物分子信息学”（biomolecular informatics）等词汇用以表示本书所关注的研究领域。为了用词灵活起见，我们在书中对于这些词汇并不加以区分，实际上读者必须注意前两个概念的范围更广，还包含免疫系统和大脑的计算机建模等本书没有讨论的研究领域。最近，计算分子生物学还被赋予了一个完全不同的含义，类似于“DNA计算”（DNA computing），这是一个用于描述利用生物分子——而不是硅片——制造计算设备的概念。本书中我们在使用神经网络的概念时，有时会在前面加上“人工的”这个形容词。这里，我们仅从模式识别算法的角度讨论人工神经网络。

最后提一句，本书所使用的大部分符号列于书的结尾处。我们一般不系统区分标量、向量和矩阵。诸如“D”这样的符号用于表示数据，但不考虑数据的复杂程度。必要时，向量都视为列向量。黑体字符通常用于表示概率概念，诸如概率（**P**）、期望（**E**）和方差（**Var**）。如果X表示一个随机变量，我们使用$\mathbf{P}(x)$代表$\mathbf{P}(X=x)$，在不产生歧义的时候，还直接为$\mathbf{P}(X)$。实际的概率分布可以记为P、Q、R等符号。

本书中，我们主要讨论离散概率分布的情况，读者也应该了解如何在必要时将结论推广到连续概率分布的情形。手写体符号用于表示特殊函数，诸如能量（$\mathcal{E}$）和熵（$\mathcal{H}$）。此外，我们还必须经常考虑用许多下标标识的变量。例如，神经网络中连接权重所依赖的其所连接的神经元i、j和所在的隐层l，在学习算法迭代中的时间t，等等。在特定的分析中，仅仅那些最具相关性的变量需要在下标中标识。在极少数不会引起歧义的地方，我们会使用相同的符号代表两种不同的意义。（例如，D也代表隐马氏模型中的删除状态。）

致 谢

多年以来，本书的工作得到了丹麦国家研究基金会和国家卫生研究院的支持。SmithKline Beecham公司对Net-ID基因片断项目的部分工作提供了赞助。本书的部分内容是皮埃尔·巴尔迪在加州理工学院生物学院时完成的。我们要向Sun Microsystems公司和加州大学欧文分校（UCI）基因组和生物信息学研究所提供的支持表示感谢。

我们要感谢所有对于手稿的早期版本提供反馈的人们，尤其是Jan Gorodkin，Henrik Nielsen，Anders Gorm Pedersen，Chris Workman，Lars Juhl Jensen，Jakob Hull Kristensen，David Ussery以及Net-ID项目的Yves Chauvin和Van Mittal-Henkle。此外还有生物序列分析中心的所有成员，他们多年来在许多方面对这项工作提供了设备上的支持。

我们还要感谢Chris Bishop，Richard Durbin和David Haussler，他们邀请我们到剑桥的伊萨克·牛顿学院，我们在那里完成了本书的第1版，我们还要感谢学院的良好环境和盛情邀请。要特别感谢Geeske de Witte，Johanne Keiding，Kristoffer Rapacki，Hans Henrik Stærfeldt和Peter Busk Laursen，他们的巨大帮助使我们将原先的手稿改变成现在这本书。

关于本书的第2版，我们要向UCI的新同事和新学生们致谢，他们是Pierre-Francois Baisnée，Lee Bardwell，Thomas Briese，Steven Hampson，G.Wesley Hatfield，Dennis Kibler，Brandon Gaut，Richard Lathrop，Ian Lipkin，Anthony Long，Larry Marsh，Galvin McLaughlin，James Nowick，Michael Pazzani，Gianluca Pollastri，Suzanne Sandmeyer，Padhraic Smyth。我们还要向以下UCI以外的人员表示感谢，他们是Russ Altman，Mark Borodovsky，Mario Blaum，Doug Brutlag，Chris Burge，Rita Casadio，Piero Fariselli，Paolo Frasconi，Larry Hunter，Emeran Mayer，Ron Meir，Burkhard Rost，Pierre Rouze，Giovanni Soda，Gary Stormo，Gill Williamson。

我们还要向本丛书的编辑 Thomas Dietterich 以及MIT出版社的工作人员表示感谢，尤其是Deborah Cantor-Adams，Ann Rae Jonas，Yasuyo Iguchi，Ori Kometani，Katherine Innis，Robert Prior。还有Harry Stanton，他在我们开始写作时提供了许多帮助。最后，我们要感谢所有的朋友以及我们的家庭所给予的帮助和支持。

本书介绍了机器学习方法的主要内容及其在生物学数据处理中的应用。其中对机器学习技术的理论基础——贝叶斯概率体系进行了详细介绍，并在此基础上着重对神经网络、隐马氏模型以及概率图模型等方法在生物信息学中的应用作了详细分析。书中特别列出一章介绍了DNA微阵列和基因表达，以及相关数据的分析方法。

本书主要针对两个读者群体，一是生物学和生物化学研究人员，他们想了解基于数据处理的算法；二是物理、数学、统计、计算机科学等领域的学者，他们想知道机器学习方法在分子生物学研究中的应用。

皮埃尔·巴尔迪(Pierre Baldi)是美国加州大学医学院信息和计算机科学系教授、生物化学系教授，基因组学和生物信息学研究所所长。

索恩·布鲁纳克(Søren Brunak)是丹麦理工大学生物系教授，生物序列分析中心主任。

责任编辑 [illegible]

封面设计 AW工作室

经销：中信联合发行有限公司

目 录

第1章　概　述

1.1　数字化符号序列中的生物学数据

与生物体功能和进化相关的链状分子具有一种基本特性，即它们能够以数字化符号序列的形式表示。DNA、RNA以及蛋白质分子中的核苷酸和氨基酸单体是确定的。虽然它们在生理环境中常常会经历复杂的化学修饰，仍然可以使用很少的字符表示其分子链的组成。因此，通过实验得到的生物序列在原则上是完全确定的。在某一序列的特定位置上，我们只能发现一种确定的单体或“字符”，而不是几种可能性的组合。

遗传数据的数字化特征使它们明显不同于其他许多科学实验数据。在其他科学实验中，物理学基本定律和实验技术的复杂性决定了实验结果或多或少有不确定性。与此相对照的是，在经济能力和其他资源允许的条件下，我们可以完全确定基因组DNA的核苷酸序列以及与之相关的蛋白质的氨基酸序列。然而，在大规模DNA测序的基因组项目中，或者进行蛋白质直接测序时，研究目标、信息检索能力、承担项目的机构、伦理和经济等因素都会影响数据质量的标准。

生物序列数据的数字化特性对算法的类型产生了深刻的影响，这些算法用于计算分析并发展得较为成熟。使用这些算法不仅能研究特定的序列及其分子结构和功能，还能常常用于对一组序列的综合研究，包括：特定序列在不同物种中的不同形式，以及在具有多态性的情况下，相同物种中序列的不同形式。为了更好地对不同物种的序列模式进行比较研究，还必须考虑到生物序列的内在“噪声”，这种噪声表现为序列片断的多样性，这种多样性部分地来自被进化放大了的随机事件。由于具有特定功能和结构的DNA和氨基酸序列存在一些不确定的差异，序

列模型必然是基于概率理论的模型。

1.1.1 数据库注释的质量

虽然由实验确定的序列数据可以达到很高的精度，但研究人员得到的数据常常包含各种噪声，这些噪声是对实验结果的错误解释和公共数据库中数据处理、存储不当等的综合结果。这似乎有些不合逻辑。但是实际上，由于生物序列都是用电子化方式存储，维护数据库的人员组成也极为不同，数据库中的数据更是由不同的生物学家和生物信息科学家提交和注释的，因此后续信息处理引人的误差要远大于初始的实验误差就不难理解了。[100,101,327]

大型序列数据库中数据的存储方式是导致这一问题的重要因素。生物序列的特征一般表现为相关位置的数字列表，而不是序列的具体内容。能够处理个人毕生积累起来的大量信息是人脑的一种重要能力，人脑是通过内容寻址（content-addressable）的存储方案来记忆信息的。通过这种方式，人脑可以根据一项记忆中很小的一部分得到这项记忆全部内容。例如，人们经常通过一首歌曲开始的两句回忆起整首歌曲。

今天的计算机是为处理数字而设计的，例如，许多国家在计算机出现以前并没有社会保险号之类的个人识别号码。[103] 计算机注释和检索信息的方式与人脑中内容寻址的步骤不同。计算机在搜索某人护照上的特征，如姓名、职业或头发颜色等的时候，并非总能挑选出惟一精确的匹配；而且在能够得到精确匹配的大多数情况下，也要求这些信息必须使用正确的语言和准确的拼写。

生物序列的检索算法可以看做根据内容的“模糊”（fuzzy）表达形式搜索特定序列的相关方法，它大大不同于根据功能对序列进行搜索。实验人员向数据库提交功能信息时，经常要将实验室中保存功能信息的序列标记、色彩或其他表示形式，转换成数据库可以接受的形式，即将这些表示具体内容的各种符号统一转换为与位置相关的整数。因此，人们不再可能直接用视觉审核这些数字形式的信息。

在序列数据库中，这种表示方法会导致一些负面后果。此时，数字特征表中的误差，而不是检索关键词可以接受的噪声，通常会导致一些垃圾信息，使得记录中序列的位置与注释的结构或功能特征之间随机地失去正确的对应关系。经常遇到的错误是基因组DNA序列中编码区或非编码区的注释错误或毫无意义，对于氨基酸序列则是功能位点或转录后修饰位点标注错误。设计一种完美的数据注释和存储方式并不容易。就目前的情况而言，生物信息学家在建立用于预测和分类的机器学习方法时，考虑这些潜在的误差来源是非常重要的。

在许多由序列决定的生物机制中，特定位置上出现某些特定的核苷酸或氨基酸是必不可少的。这类先验知识有助于发现数据中的“印刷”错误（typographical error）。有趣的是，机器学习技术提供了另外一种非常有效的发现错误信息和错误注释的方法。在一批数据中，如果某些样本难以学习，那么它们很有可能是某些非正常情况，或者是功能注释不正确。在这两种情况下，最好能够剔除显著背离一般规律的样本。通过这种方式，机器学习技术已经发现了真核生物基因内含子剪接位点的标识错误，[100,97,101,98,327]哺乳动物蛋白质中O联糖基化位点的标识错误或遗漏，[235]小RNA病毒中多蛋白剪切位点的错误标识，[75]等等。值得注意的是，并非所有的误差都来源于数据处理，例如，将论文中的信息转化为数据库数据项的过程中也会引入误差——相当大量的误差来源于实验人员所做的错误标识。[327]许多这类错误可以在加入公共数据库之前由简单的一致性检验发现。

在公共数据库中，特征注释原始依据的不确定性是一个很普遍的问题。注释的依据可能是实验方法，也可能是序列的相似性，或者仅仅是某个预测算法。在数据库中，这种情况经常导致一些较为随意的标识，使用诸如问号（？）或“有可能的”（POTENTIAL）、“大概的”（PROBABLE）等标注方式，不便于自动分析。为了避免对特定算法的预测性进行循环评价，需要对数据进行精心的准备并且删除来源不明的数据。如果没有正确使用数据，随着更多预测算法的出现，这个问题在将来会变得越来越严重。机器学习技术之所以能够在这样一个数据并不精确的领域中获得成功，原因之一在于：与相对应的生物学机制类似，它在庞大的序列数据集支持下可以很好地处理数据中的噪声。在相关的自然语言学习研究中，最新发现表明：8个月的婴儿就能够发现一些语言规律，并且通过学习简单的统计特征识别连续语音中的词语边界。[458]语言学习对于婴儿正如DNA序列对于我们一样神秘和复杂，因而机器学习技术自然也能有助于揭示基因组数据中的类似规律。

1.1.2 数据库冗余

在蛋白质和DNA序列分析中，另一个经常困扰研究者的问题是数据冗余。蛋白质或基因组数据库中的许多记录代表一些蛋白质和基因家族的不同成员，或者在不同生物中发现的同源基因的不同版本。一些研究小组可能分别提交了相同的序列数据，因此相对应的数据项即使不是完全相同，也多少是密切相关的。这些相似序列的注释在最好的情况下也只是大致相同，而某些显著的区别可能反映了生物体或组织真正的特异性。

在测序项目中，典型的数据冗余来自不同的实验手段本身。例如，一条特定

的DNA序列片段既可以通过基因组形式测序，还可以通过cDNA的形式测序，因为cDNA是与细胞中转录的RNA序列互补的。由于存入数据库的序列是以广泛多样的手段获得的——从包含噪声的单遍测序结果到通过5~10遍重复测序获得的完整数据，数据库中的不同记录可表示同一个基因，但是这些记录之间可能存在不同程度的差异。

在大量真核生物中，cDNA序列（完整的或不完整的）代表了mRNA前体剪接后的形式，这意味着：对于那些经过可变剪接的基因，其基因组中DNA片段一般对应于几条在染色体中并不连续的cDNA序列。[501]可变剪接的产生可以有多种不同的方式。图1-1给出了一些剪接过程中编码区和非编码区结合、跳过或替换的不同方式。同样利用剪接机制的生物，在进行可变剪接时表现得非常不同。显然，可变剪接机制的替代方法是在基因组中直接包含相同基因的一些不同版本。这可能正是秀丽线虫（*Caenorhabditis elegans*）所采用的策略，它可能包含大量非常相似的基因，这在转化为数据集时也产生了冗余。[315]在人类基因组中，至少30%~80%的基因可能具有可变剪接机制。[234,516,142]事实上，可变剪接可能是一条准则而不只是一种特例。

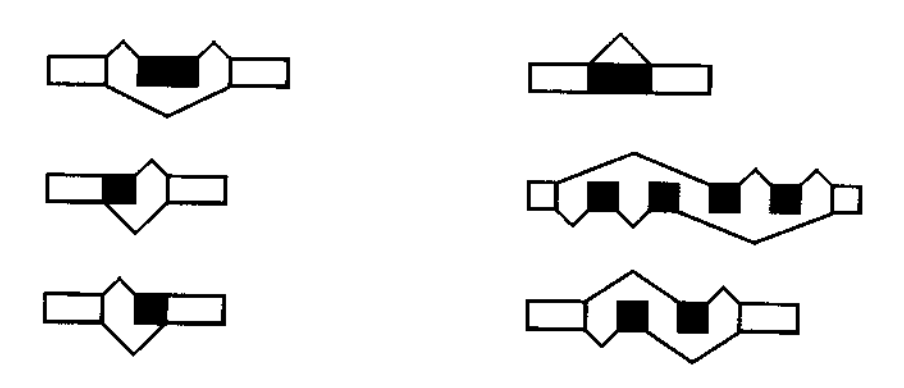


图1-1　真核生物可变剪接最普遍的一些模型

左列从上向下依次为：外显子序列盒（跳过或包含外显子），可变5'剪接位点，可变3'剪接位点。右列从上向下分别是：保留整个内含子，成对剪接的外显子以及外显子的互斥。这些不同类型的mRNA前体的可变处理过程可以进行组合。[332]

数据冗余还可能对大规模并行基因表达实验产生重要影响，我们将在第12章进一步讨论这个问题。无论是被点样到玻璃平板上，还是直接在DNA微阵列上生成，实验中所使用的基因序列总是基于存储在数据库中的序列或序列族。这样，微阵列中最终包括的序列，会多于特定生物体全基因组中的基因数目，从而在定

量化芯片实验的杂交记录时产生噪声。

在蛋白质数据库中，特定基因也可能表示为某种氨基酸序列，但该序列并不对应于原始核苷酸序列的直接翻译。例如，蛋白质序列经常需要通过微小的修饰，以便获得更好的晶体，利于采用X射线晶体衍射法测定结构。[99]氨基酸的删除和替换也是导致数据冗余的常见原因。

使用具有冗余的数据集至少会导致三种误差：第一，如果数据集中的氨基酸或核苷酸序列包含很大的密切相关的序列家族，统计分析将偏向这些家族，并侧重描述它们具有的特征；第二，序列不同位置之间表面上的相关性可能是对序列数据进行有偏倚的采样所导致的人为特征；第三，在我们使用数据集对某一特征进行预测或用于选择、标定预测方法时，如果用于训练和标定预测方法的训练集的数据与用于测试的序列相关性过于密切，显然会过高估计预测方法的性能。这样得到的性能评估只反映出该方法重现特定输入的能力，而不代表该方法具普遍性。

当训练集中某些类型的序列样本数目过多时，至少某些机器学习方法会遇到麻烦。尽管已经有了解决这一问题的算法，但是首先对数据集进行整理，使低显现度（underrepresentation）的序列得到均等的机会，经常会获得更好的效果。重要的是要意识到低显现度会同时在模型的基本结构层次（序列冗余）和分类层次上造成困难。例如，由于在蛋白质二级结构中，无规卷曲比β折叠更为常见，因此在进行分类时预测结果会偏向于无规卷曲。

由于这些原因，数据集需要尽量避免包含过于密切相关的序列。另一方面，对于“过于密切相关”太精确的定义又可能导致丢弃数据集中有价值的信息。因此，我们必须在数据集的规模和非冗余之间寻求折中。恰当地定义“过于密切相关”很大程度上依赖于所解决的问题。然而在实际操作中，人们却很少考虑到这个问题。人们经常称测试数据是从全数据集中“随机”选取的，暗示数据是经过精心准备的，而实际上却完全没有进行降低数据冗余的工作。而即使采用了降低数据冗余的方法，在大多数情况下，其做法或者是采用了多少有些随意的相似度阈值，或者只是根据传统的蛋白质和基因家族列表，在每个家族中选择一个成员从而构造出一个“代表”数据集。

另一种替代策略是保留数据集中的全部序列，根据序列的奇异程度赋予它们不同的权重。对于密切相关序列的预测将得到很低的分值，而相关距离较远的序列则构成了预测的主体。这一方法的主要风险在于错误数据总是与较大的权重相关联。通常我们可以辨别出注释错误的序列，至少当误差来自于数据库中特征表的“印刷”错误时。但对于赋值错误的特征所做出的预测将影响到对整个模型的评价，甚至可能导致对预测性能的严重低估。不仅假位点很难预测，而且那些可

能在正确注释中出现的真位点也经常被计为假阳性。

序列谱（sequence profile）是一种利用数据库冗余的非常有效的方法，[226]它不仅与基于比对的序列检索有关，而且关系到机器学习算法中如何设计输入数据的表示方法。序列谱描述了通过多重序列比对组织起来的一族序列中每个位置上氨基酸的变化。由于序列谱中不再包含关于单独序列模式的信息，其中序列变化程序的信息在数据库搜索中是极为有用的。在类似PSI-BLAST的程序中，可以根据当前版本的序列谱选出的序列，反复更新序列谱。[12]在后续章节中，我们将讨论隐马氏模型，该模型以一种非常灵活的方式贯彻了序列谱的概念。此外，神经网络也可以接受序列谱信息作为输入数据。所有这些方法都利用了存储在公共数据库中的信息冗余。

1.2 基因组——多样性、规模和结构

生物体的基因组具有广泛的多样性。这种多样性不仅包括基因组的规模还包括基因组的存储方式（使用单链或者双链的DNA或RNA）。另外，有些基因组是线形结构（如哺乳动物），有些则是封闭的环形结构（如大部分细菌）。

细胞的基因组都是由DNA组成的，[389]而噬菌体和病毒的基因组则可能由DNA或RNA组成。在单链的基因组中，信息可以从正向或从反向读取，还可以从两个方向同时读取，这种情况下，我们称其为双义基因组（ambisense genome）。其中，"正向"定义为从分子的5'端到3'端。在双链的基因组中，只从正向读取信息（从每条链的5'端到3'端）。并不是所有生物的基因组都采用直接复制的方式，例如：逆转录病毒的基因组由RNA组成，然而在复制时需要利用一种由DNA构成的中间产物。

不能自我复制的微生物如噬菌体和病毒等，具有最小的基因组，它们分别利用寄生的原核或真核生物的代谢和复制机制生存。1977年，科学家首先对噬菌体ϕX174基因组中长为5 386bp的序列进行了测序。[463]这么小的基因组一般只具有1条染色体。然而有些小基因组却具有多条染色体，例如在1996年完成测序工作的嗜热古细菌（*Methanococcus jannaschii*），其基因组大小仅为1.74Mbp。嗜热古细菌具有3条染色体，其中1条比另2条大得多。而更大规模的总长为3 310Mbp的人类基因组则由22条常染色体和2条性染色体组成。即使同是灵长类动物，染色体数目也存在差异。如黑猩猩就有23条常染色体和2条性染色体，因此黑猩猩体细胞核中总共有48条染色体，而人类体细胞核中只有46条。其余的哺乳动物具有完全不同的染色体数，如猫有38条染色体，狗的染色体则有78条之多。由于大部

分高等生物的DNA具有两份近似的拷贝（二倍体基因组），因此我们在讨论其中的一份拷贝时，也使用DNA单倍体的概念。

某些生物的染色体是不稳定的。例如，研究人员发现*Bacillus cereus*的染色体由两部分组成，较长的一部分（2.4Mbp）较为稳定，而较短的一部分（1.2Mbp）不够稳定，很容易插入大达兆数量级的长度不同的外染色体元件。[114] 这点会给测定这个生物的基因组序列或得到遗传图谱造成很大的困难。几乎所有基因组转座元件（transposable element）都会造成相当长的序列重排或插入，尽管还没有发现它们能够改变染色体的数目。一些理论认为有很大一部分染色体的组成具有优越性，可以加快进化的速度，然而关于这个问题目前没有定论。[438]

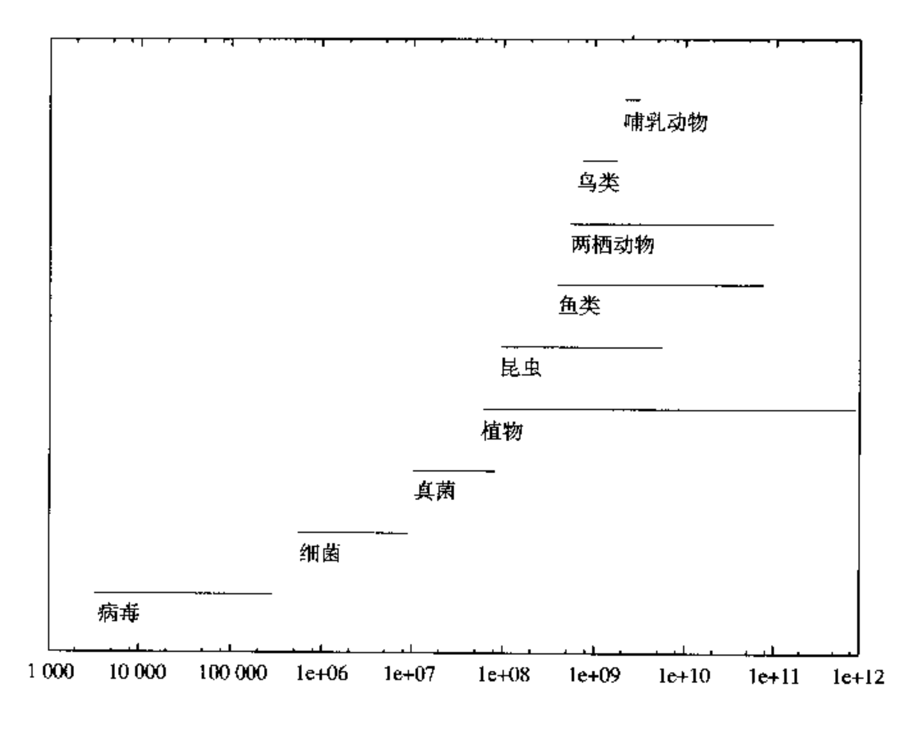


图1–2 不同类型生物的基因组大小区间

注意图上横轴的坐标是碱基数目的对数值。组内的差异一般都大于1个数量级，哺乳动物基因组大小的区间非常狭小，是这一个例外。一种很自然的想法是图的纵轴代表生物的复杂性，但是大部分情况下纵轴并不直接代表基因库的大小。处于图上方的许多生物，如哺乳动物、鱼类和植物，它们的基因数目相差无几（见表1–1）。

一个有趣的现象是基因组大小的分布图在某种程度上分隔成一些互不重叠的区间。图1–2表明，病毒的基因组大小在3.5Kbp~280Kbp之间，细菌的基因组则

在0.5Kbp~10Mbp之间，真菌是10Kbp~50MbP，植物的基因组最小在50Mbp左右，而哺乳动物的基因组规模则位于1Gb附近的一个较窄的区间上（以对数尺度考虑）。生物体具有不同的生命形式，如非细胞形式（病毒），单细胞形式（细菌），不具有复杂的细胞间通信的多细胞形式（真菌），具有许多细胞间信号转导系统的高等分化的多细胞形式（哺乳动物和植物）。图中这种阶梯形结构反映了维持不同生命形式所必需的基因库的规模。近几年的研究表明甚至细菌也能通过化学信号进行通信。[300]细胞“通信员”能在分子间往返，实现种群规模的控制。一个典型的例子是荧光素酶的基因表达，它和其他蛋白质一起参与海洋细菌的荧光反应。当然，与高等生物的信号转导相比，这种形式的通信只需要较为有限的基因库。

大部分类别中，基因组的规模差异较大。在真核生物中，有一些类别则是例外（如哺乳动物、鸟类和爬行动物），其基因组规模局限在一个较窄的范围。[116]由于能通过染色体光学作图（optical mapping）等方法估计未测距部分的长度，我们可以得到人类基因组规模较为精确的估计。表1-2列出了人类24条染色体的估计长度。人类基因组序列总共大概包含3 310 004 815个碱基对，这个估计大概在一段时间内不会有太大的变化。

不同种类生物细胞中的DNA含量相差最大可以达到百万倍。细菌基因组的规模大致与其遗传和机体复杂性相关，而某些真核生物的DNA含量甚至会超过基本蛋白质编码所需DNA含量的50 000倍。[116]分子基本构造大致相同的生物，其基因组规模却会有很大的差别。脊椎动物有许多相似的生命机能，然而它们的基因组规模却非常不同。早在1968年，研究人员已经发现包括河豚在内的某些鱼类，尤其是Tetraodontidae家族的基因组规模很小。[254,92,163,534,526]河豚基因组的单倍体DNA含量在400Mbp~500Mbp的范围，而人类3 310Mbp的基因组差不多是它的6~8倍。河豚（*Fugu rubripes*）的基因组仅为最简单的秀丽线虫（100Mbp）的4倍，是人类基因组的1/8。单个细胞具有最大DNA含量的脊椎动物是两栖动物，它们的基因组大小跨度很大，从700Mbp~80 000Mbp。尽管如此，它们在结构和行为上显然没有人类复杂。[365]

表1-1　不同进化谱系生物基因的大概数目和基因组规模

分　组	物　种	基因数目	基因组规模
噬菌体	噬菌体MS2	4	0.003 569
	噬菌体T4	270	0.168 899
病毒	花椰菜花叶病毒	8	0.008 016
	HIV-2病毒	9	0.009 671
	痘苗病毒	260	0.191 737

（续表）

分 组	物 种	基因数目	基因组规模
细菌	生殖道支原体	473	0.58
	肺炎支原体	716	0.82
	嗜血流感杆菌	1 760	1.83
	枯草杆菌	3 700	4.2
	大肠杆菌	4 100	4.7
	黄粘液球菌	8 000	9.45
古细菌	嗜热古细菌	1 735	1.74
真菌	酿酒酵母	5 800	12.1
Protoctista	*Cyanidioschyzon merolae*	5 000	11.7
	Oxytricha similis	12 000	600
节肢动物门	黑腹果蝇	15 000	180
线虫纲	秀丽线虫	19 000	100
软体动物门	*Loligo pealii*	20-30 000	2 700
植物界	烟草	20-30 000	4 500
	拟南芥	25 500	125
脊索动物门	*Giona intestinalis*	N	165
	河豚	30-40 000	400
	Danio rerio	N	1 900
	小家鼠	30-40 000	3 300
	智人	30-40 000	3 310

基因组大小以Mbp为单位。"N"代表没有确切值。部分数据来自参考文献［390］及其中提到的参考文献（根据最新的估计重新计算），其余的来自一系列网络资源、论文和书籍。

表1-2　人类基因组序列24条染色体的近似大小

人类染色体	大 小
1号染色体	282 193 664
2号染色体	253 256 583
3号染色体	227 524 578
4号染色体	202 328 347
5号染色体	203 085 532
6号染色体	182 415 242
7号染色体	166 623 906
8号染色体	152 776 421
9号染色体	142 271 444
10号染色体	145 589 288
11号染色体	150 783 553
12号染色体	144 282 489
13号染色体	119 744 898

（续表）

人类染色体	大 小
14号染色体	106 953 321
15号染色体	101 380 521
16号染色体	104 298 331
17号染色体	89 504 553
18号染色体	86 677 548
19号染色体	74 962 845
20号染色体	66 668 005
21号染色体	44 907 570
22号染色体	47 662 662
X染色体	162 599 930
Y染色体	51 513 584

值得注意的是其中22号染色体的大小排位与染色体的编号并不一致。数据来自Ensembl（www.ensembl.org）和Santa Cruz（genome.ucsc.edu）网站。人类基因组序列总共大概包括3 310 004 815个碱基对，这个估计大概在一段时间内变化不大。

1.2.1 人类基因组和其他基因组的基因容量

生物体全基因组序列的不同部分包含有基因。基因这个概念通常定义为可表达单元的一个或多个序列片段。"基因"这个单词（与"基因型"（genetype）和"表现型"（phenotype）这些概念一起）由丹麦遗传学家威廉·约翰森（Wilhelm Johannsen）于1909年创造，这比人们详细了解DNA的物质基础早得多。

基因可以编码蛋白质产物，或者编码多种RNA分子中的一种，这些RNA分子对于细胞处理遗传物质和正确执行功能是必需的。细胞质内的mRNA序列在制造相同蛋白质的多份拷贝时，可以作为复制模板。编码其他RNA分子的基因必须转录一定的数量。不直接导致基因产物的序列片段通常称为非编码区。非编码区可以是基因的一部分，也可以是基因的调控元件或者间插序列，后者打断了直接编码蛋白质或RNA的DNA序列。机器学习技术对于解释尚未注释的基因组的DNA序列以及辨别具有不同功能的序列之类的困难任务，是极为理想的。

表1-1是当前对于不同进化谱系（evolutionary lineage）生物的基因的大致数目和基因组大小的预测。对于那些已经进行全基因组序列测序的生物，这些数目当然是非常精确的；而对于其余生物，我们只能得到基因密度的粗略估计。像细菌这样的生物，其基因组大小是一个很强的生长抑制因素，所以编码区（编码蛋白质和RNA）几乎覆盖了整个基因组；而对于其他生长缓慢的生物，编码区仅占了全基因组的1%~2%。由于计算方法在进行基因发现时需要利用基因密度，这意

味着基因密度常会对算法的精度产生很大影响。基因组的非编码区通常包括许多伪基因（pseudo-gene）以及其他序列，它们在用算法进行全基因组扫描时会呈假阳性状态。

表1-3 20个测序完毕的常见生物的碱基数目（2001年4月，GenBank的第123版）

物　种	单倍体基因组规模	碱基数目	记录数
智人	3 310 000 000	7 387 490 518	4 544 962
小家鼠	3 300 000 000	1 527 228 639	2 793 543
黑腹果蝇	180 000 000	502 655 942	167 687
拟南芥	125 000 000	249 689 164	183 987
秀丽线虫	100 000 000	204 396 881	114 744
Oryza sativa	400 000 000	171 870 798	161 411
黑绿色河豚	350 000 000	165 542 107	189 000
褐鼠	2 900 000 000	114 331 466	229 838
牛	3 600 000 000	76 700 774	168 469
Glycine max	1 115 000 000	73 450 470	167 090
平头苜蓿	400 000 000	60 606 228	120 670
番茄	655 000 000	56 462 749	109 913
布氏锥虫	35 000 000	50 723 464	91 360
普通大麦	5 000 000 000	49 770 458	70 317
兰氏贾第鞭毛虫	12 000 000	49 431 105	56 451
紫色球海胆	900 000 000	47 633 412	77 554
Danio rerio	1 900 000 000	47 584 911	93 141
爪蟾	3 100 000 000	46 517 145	92 041
玉米	5 000 000 000	45 978 459	98 818
痢疾阿米巴	20 000 000	44 552 032	49 969

由于菌株的不同或者纯粹的冗余，某些生物的序列长度远比所列出的基因组大小更大。

最令人惊奇的结果来自于对于人类基因组两个不同版本数据[134,170]的分析，科学家发现其中所包含的基因数量仅在30 000这个数量级。序列的初步分析仅估计出30 000~40 000个基因。当然，这并非完全出乎意料，因为果蝇的基因数目（14 000）也出人意料的少。[132]但是人类的基因数目不足简单的秀丽线虫的2倍，如何实现其复杂的生物功能？答案部分来自于这些数目有限的基因的可变剪接以及其他实现基因多功能化的方式。这个领域过去在基础研究中没有得到应有的重视，而人类基因组的研究成果发布清楚表明了我们过去的无知；仅在人类基因组数据公布1年前，人们还估计其中大约有100 000~120 000个基因。[361]对于一个复杂生物，基因多功能化使基因组中许多基因都可以制造几个不同的转录本，而

每个转录本还可以产生多种蛋白质变体。由于遗传物质的细胞处理过程在调控方面远比原先设想的复杂，人们更为迫切地需要能够对这些过程建模的更加高级的生物信息学方法。

一个尚未解决的重大问题无疑是基因库规模微小的增加如何导致生物复杂性的显著提高。秀丽线虫的基因数目和人类几乎差不多这个事实令人可气。在以完整细胞和完整生物体作为研究对象的时代，我们需要了解基因组中固定数目的基因可能以什么方式决定生物的复杂性。

法国生物学家让-米歇尔·克拉弗里（Jean-Michel Claverie）对于生物复杂性K和基因组中基因数目N之间的关系进行了一个有趣的"个人"估计。[132] 将N映射成K的函数 f 原则上可以是线性函数（$K\sim N$）、多项式函数（$K\sim N^a$）、指数函数（$K\sim a^N$）和阶乘函数（$K\sim N!$）等。克拉弗里提出复杂性应该和生物体的基因表达多样性的能力，也就是其所能实现的理论上的转录组状态的数目相关联。对于最简单的模型，假定基因只有激活或者失活（"开"或者"关"）两种状态，具有N个基因的基因组总共可能编码2^N种状态。如果我们把人类和秀丽线虫进行比较，我们的复杂程度将是秀丽线虫的

$$2^{30\,000}/2^{20\,000}\cong 10^{3\,000} \tag{1.1}$$

倍，这点增强了（也许是重建了）我们关于人类优越性的主观看法。在这个简单模型中，指数值显然必须下调，因为基因冗余或共调控使基因的表达相互并不独立，而且许多状态实际上是致命的。另一方面，基因表达并不是简单的开/关，而是以一种更具层次性的方式进行调控。一个非常简单的数学模型可以说明为何基因数目的略微增加可以导致复杂性的显著提高，并提出了解决全基因组测序带来的N值悖论的一种方法。这个基于基因表达模式的模型看上去非常简单，它仍然代表了对于生物的"系统"特性进行定量化的一种尝试，尽管其构成只能从传统的还原论方法去理解。[132]

另外一个十分基础而且在很大程度上还没有解决的问题是，为什么在许多高等生物的基因组中，编码蛋白质的部分非常有限。在人类基因组序列中，无论我们对基因数目N使用较为悲观的估计（26 000）或是乐观的估计（40 000），编码区域所占的比例都很小。[170] 根据这两个估计，人类基因组序列似乎仅有1.1%（1.4%）是编码区域。相应地，内含子覆盖了25%（36%）的区域，而基因之间的剩余部分占据了75%（64%）。人们常认为基因仅占据了几个百分点的区域，这显然是不对的，因为人类的内含子具有较大的平均长度。以基因数为40 000做估计，基因覆盖了人类基因组超过1/3的区域。

对于特定生物，其尚未复制的单倍体基因组中核DNA的质量称为C值，因为在限定范围较窄的生物中，这个数值是一个常数。真核生物基因组的C值由于物种的不同最多可以相差80 000倍，然而这与生物复杂性或者编码蛋白质的基因数目之间却没有什么关系，[412,545]这种现象称为C值悖论。[518]

有人提出，非编码的DNA序列在核内基因组中的累积受其复制所需的成本的限制，该成本超过仅在核中起结构支撑作用所需成本。[412]在许多年前人们已经知道外显子之外的DNA序列通常并不增加基因的数量。如果大容量的基因组中每个基因只是按比例增加拷贝数目，则DNA复性实验（DNA renaturation experiment）的动力学过程会非常快。在复性实验中，对加热失活的DNA样本进行冷却，如果DNA链之间具有充分的互补，它们会重新结合。而实验显示这个动力学过程相当慢，这表明大容量基因组中的外显子之外的DNA不太可能编码基因。[116]研究人员在植物中发现了一些大小相差非常离谱的基因组，而现在有明显的证据表明基因组大小和气候之间存在一定的关联，[116]而且基因组大小的明显差异还必须从分子和进化机制的角度进行解释。有人提出，无论如何，基因组的信息总量并不能很好地提示基因组的“质量”及效率。

这种情况也许并没有看上去的那么奇怪。实际上，与人类之间的交流相似，消息长度并不能很好地代表所交换信息的质量。例如，在科学文献或者合作者之间的通信中，简短的交流可以非常有效。许多电子邮件省略了很大一部分繁文缛节，只以非常紧凑的形式留下要点。我们所知的世界上最短的信件是极端有效的：1862年《悲惨世界》出版之后，维克多·雨果出去度假，但是他非常渴望知道书的销量如何。他给出版商写了一封信，上面仅仅是一个“？”。出版商回信，也仅仅是一个“!”。雨果看了以后就毫无顾虑地继续他的度假。那本书成了畅销书，至今仍然很受欢迎，其所改编的音乐剧和电影也是如此。

图1–3显示出GenBank数据库[62,503]规模的指数增长状况。表1–3列出了20个完成测序的常见生物。由于几年来这种数据一直以相同的速度指数增长，因此在新的、更快的甚至更便宜的测序技术出现以前，我们很容易在这张图上进行外推。如果发明出更新的测序方法，这种增长的速度大概会加快。否则，这种增长速度可能停滞不前，因为一些哺乳动物基因组已经测序完毕。如果那时测序依然花费巨大，投资机构可能会把资源转向其他科研领域，导致更低的数据增长率。

除了存储在GenBank中可供公众利用的数据外，许多公司和其他机构的私有数据也以很快的速度增加，因此很难估计当前人类已知的序列数据的总数。今天，在一些最大的公司，并行操作几百个测序机器对相同染色体的不同区域

进行测序，对原核生物全基因组的原始测序可以在一天之内完成。这些数据的一部分将最终存放在公共数据库，而其余的依然保留在私有数据库中。对于所有生物体而言，提高测序的速度十分重要，这并不仅仅因为序列数据的产生是与专利相关的。

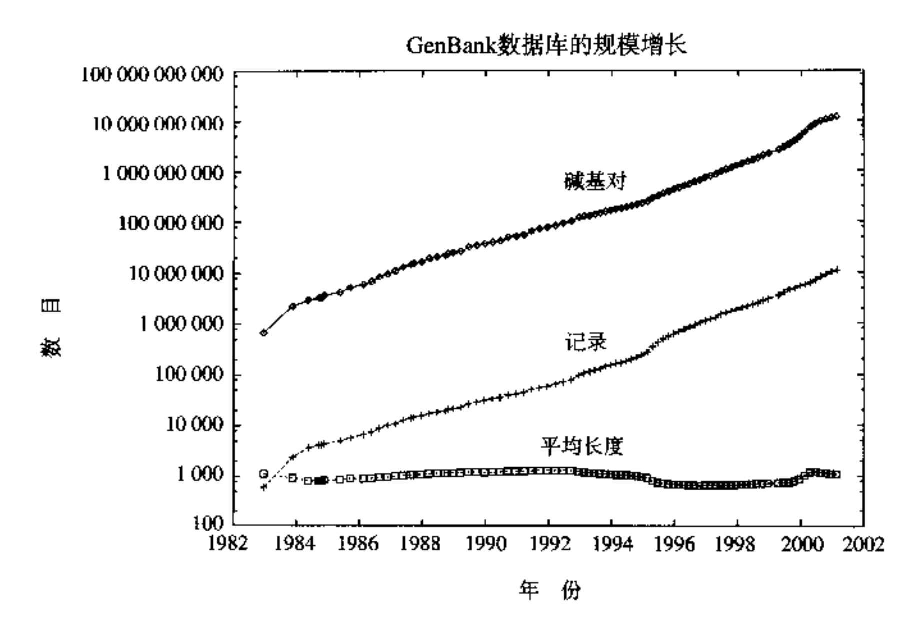


图1-3 1983年至2001年期间，GenBank数据库规模的指数形式增长

根据2000年至2001年的增长情况，倍增时间为10个月左右。GenBank第123版的完整大小为11 545 572条记录，共12 418 544 023个核苷酸（平均长度为1 076）。现在这个数据库每天的增长量超过11 000 000个碱基。

1.3 蛋白质和蛋白质组

1.3.1 从基因组到蛋白质组

在蛋白质层次上，与全基因组（complete genome）的大规模分析相对应的研究称为蛋白质组（proteome）分析。[299,413]蛋白质组包括一组染色体所有蛋白质的表达。在多细胞生物中，所表达的蛋白质根据细胞类型而有所差异。蛋白质表达也会随时间变化，这是由于基因调控会从胚胎阶段开始随着发育阶段的不同而不断改变。蛋白质组的研究对象是特定基因组中的基因所产生的蛋白质。

“基因组”这个术语是由德国植物学家汉斯·温克勒（Hans Winkler）在第一次

世界大战前创造的，[561,65]而“蛋白质组”直到最近才出现在科学文献中，由马克·威尔金斯（Marc Wilkins）和基思·威廉斯（Keith Williams）在1994年首次提出。[559]

蛋白质组分析的内容不仅包括编码蛋白质的基因的序列、位置和功能，更注重于以转录后修饰形式存在的每一个蛋白质的准确生化状态。许多例子表明，可以应用机器学习技术成功地预测蛋白质的活性和功能。

蛋白质通常需要经历一系列的修饰以改变其活性。例如，某些特定的氨基酸会通过共价键（或者非共价键）与碳水化合物相连，这些氨基酸称为糖基化位点（glycosylation site）。有些氨基酸则会经历磷酸化（phosphorylation），即磷酸（phosphate group）通过这些氨基酸与多肽链相连。在以上两种情况中，蛋白质通过特定的一系列酶催化发生改变，它们对于蛋白质执行功能都是必需的。还存在其他类型的转录后修饰，例如分泌性蛋白转座通过细胞膜时，需要加入脂肪酸和剪切N端的信号肽。蛋白质的这些修饰作用，对于数据驱动的预测研究很有意义，因为在公共数据库中，存在很大一部分经实验验证的功能位点和序列。

1.3.2 蛋白质长度分布

生物在进化过程中，选择那些能够在水或脂环境中保持稳定构象的多肽链，这些多肽链在这样的环境下执行功能。一个重要的事实是，氨基酸序列中相距较远的残基的相互作用在蛋白质折叠中起关键作用。这种长程效应也正是利用计算方法预测蛋白质折叠的主要障碍。因此，这个方面的研究主要还是集中在局部结构预测。用于与基于分子作用力和动力学方程的计算方法一样，预测和分类方法也主要用于分析局部结构。

从伊卡斯（Ycas）和盖莫（Gamow）的早期研究开始，统计分析在蛋白质序列和进化研究中一直起着很重要的作用。[195,575,555]以往工作主要集中在具有特定结构或者功能的局域非随机模式的统计关系，现在随着数据的大量涌现，已经可以得到关于全基因组的可靠的全局统计结果。

蛋白质序列数据既可以在不同物种中进行分析，也可以从特定生物体的角度加以研究。例如，我们完全可以确定可能的最大蛋白质组中多肽链的长度分布。这里的关键问题是我们今天所看到的蛋白质序列是否代表了早期序列的一些主要特征。那些早期序列的组成基本是随机的，经过漫长的进化才形成了我们现在所看到的序列形式。[555]另一种可能性则是这些序列在早先产生的时候，其组成就具有相当大的偏倚。

利用可溶性蛋白质当前的氨基酸组成，我们可以构造出规模为10^{112}数量级的，

长度为100个氨基酸的“自然”序列。自然界只使用了这些可能序列中很小的一部分。怀特（White）和雅各布斯（Jacobs）提出过一种“随机起源假设”，该假设认为这些蛋白质来源于由某些简单规则指导的随机过程。[556,555]形式上这种理论可以在检验蛋白质序列随机性的不同方面时作为无效假设，尤其是可用于考察实际的蛋白质序列在多大程度上与随机序列相区分。

关于蛋白质一级结构在较长区域内存在一定规律性的证据也越来越多。令人惊奇的是，甚至在序列组成这个层次以下，还存在某种与物种相关的规律性：原核生物蛋白质的典型长度与真核生物蛋白质的典型长度完全不同。[64]真核生物蛋白质折叠成紧凑结构的可能性随着序列长度的增加而增加比原核生物更快，这与上述现象可能有联系。[555]有人认为我们所观察到的序列长度的差异可以用二硫键集中程度不同及其对于最优结构域大小的影响加以解释。[304]

人们还研究了其他一些类型的较长区域范围内的规律性，例如，在β折叠[543,570,268,45]和紧密接触对（close contact pair）[273]中对于某些相同或相似模式的偏好，堆积密度（packing density）的长程和短程周期性，[175]氨基酸序列中的突变是否在长程上显著相关等。[515,485,214]

原核生物与真核生物全基因组序列的出现，使我们可以在比较全基因组时检验早期基于一些不完整或冗余数据所做的观察是否正确。观察到的一个很令人惊讶的现象是，蛋白质似乎由不同的序列单元构成，真核生物中这些序列单元的特征长度约为125个氨基酸，在原核生物中则为150个氨基酸。[64]这表明蛋白质序列组织可能具有某种内在的规律，这种规律比序列自身更具研究的基础性。如果这种组织规律确定是通过进化产生的，那么多肽链的长度分布可能会取代传统上所认为的蛋白质的“一级”结构成为研究基础。

1995年，第一个在自然环境下生存的生物体——原核生物嗜血流感杆菌（*Haemophilus influenzae*）——的完整基因组公布，供研究人员分析。[183]这个环形的基因组大小为1 830 137bp，包括1 743个预测的蛋白质编码区以及76个编码RNA分子的基因。图1-4给出了这个微生物所有的预测的蛋白质长度分布。为了进行比较，图中还给出了生殖道支原体（*Mycoplasma genitalium*）全基因组中大约468个蛋白质的长度分布，[189]以及嗜热古细菌全基因组中大约1 735个预测的蛋白质编码区的长度分布。[105]

通过这些原核生物长度分布与真核生物酿酒酵母（*Saccharomyces cerevisiae*）（图1-4）的比较，可以得到原核生物（嗜血流感杆菌）与真核生物（酿酒酵母）分布的峰值显然位于不同的区间：分别是140~160和100~120。

结合冗余性约简与谱分析可以得出结论：很大一部分真核生物的蛋白质长度分

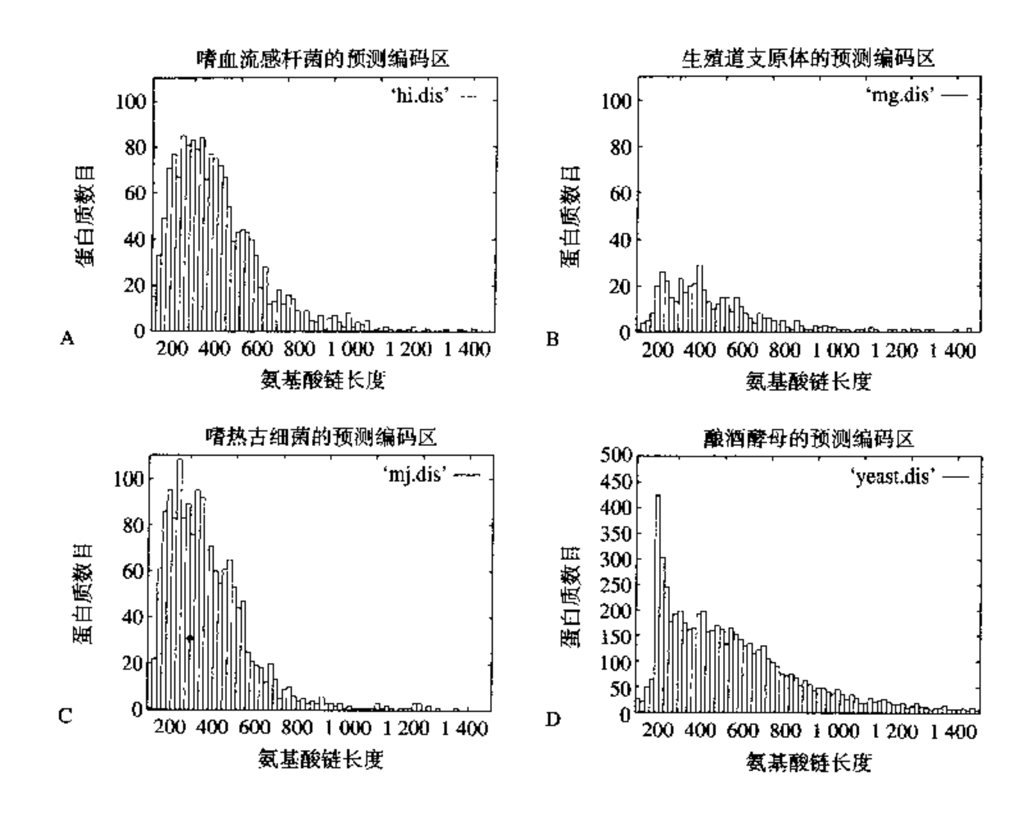


图1-4 全基因组中预测的蛋白质编码区的长度分布

A. 嗜血流感杆菌，在1 743个区域中，长度为140~160的氨基酸链出现得最为频繁。B. 生殖道支原体，共有468个区域，首选的氨基酸链长度为120~140或280~300。C. 嗜热古细菌具有1 735个区域，长度为140~160的氨基酸链出现得最多。D. 酿酒酵母，在6 200个区域中，长度为100~120的氨基酸链出现得最为频繁，其次是120~140的区间。根据1997年《自然》杂志（*Nature*）上一篇通讯，酿酒酵母中长度为100~120的序列（很可能是一些虚假样本）多得超过预计。[144]

布峰值位于125个氨基酸，并且其分布显示出以这个尺寸为周期的周期分布。[64]图1-4D也清楚地显示出长度分布较弱的第二个和第三个峰值位于210和330个氨基酸左右。这里的分布曲线是基于这个生物所有的蛋白质数据记录，并没有经过冗余性约简。

有趣的是，嗜热古细菌的分布曲线位于嗜血流感杆菌和酿酒酵母的分布曲线之间。这和一个新提出的理论相吻合，这种理论认为古生物界的物种与真核生物具有许多共同点，而并非原核生物界中一种特殊类型的细菌。[564,105,197]这表明全体始祖生物决定了细菌、古细菌和真核生物中的保守特征：

$$\text{原核生物（没有细胞核）} \neq \text{细菌} \quad (1.2)$$

古生物分类问题已经在一些教科书和序列数据库中造成了生物分类的混淆。[197]

公共数据库中经过注释的蛋白质一级结构数据也增加得很迅速。表1-4显示了SWISS-PROT蛋白质序列数据库[24]中一些重要生物的蛋白质的序列数目。从图1-5我们还可以看到数据库规模的变化。与GenBank数据库一样，这个数据库也以指数速度增长，不过相对慢得多。这说明对于预测出的基因给出具有生物学意义的阐述，这项研究进展很缓慢。对于DNA测序工程中得到的信息进行功能注释尤其需要新的技术。[513]

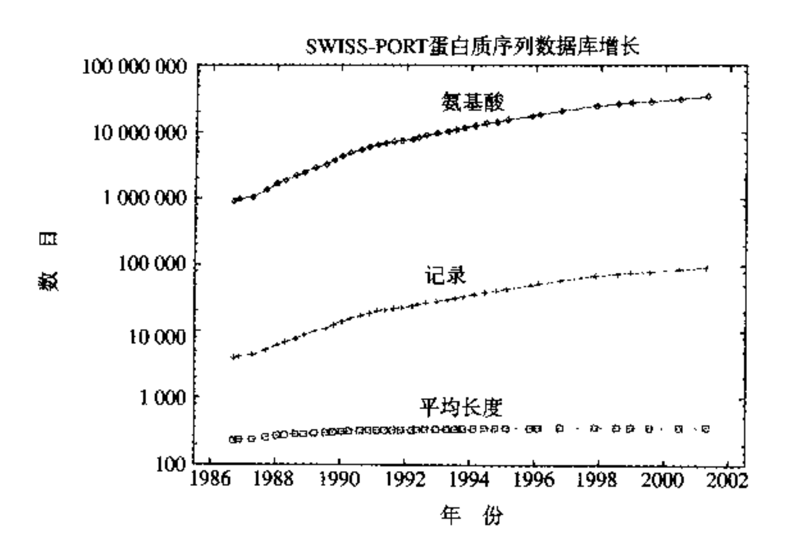


图1-5　1987~2001年期间，SWISS-PROT数据库的指数增长

SWISS-PROT数据库39.16版本中，大约有95 000条记录和34 800 000个氨基酸。

另外一个增长速度更慢的数据库是PDB（Protein Data Bank）数据库。这反映了无论使用X射线结晶或NMR技术来确定蛋白质的三维结构，实验工作仍然相当艰巨。然而从图1-6可以看出，这个数据库也是以指数速度增长，而且由于美国、日本和欧洲的许多结构基因组项目启动，它在很长一段时间内仍然会保持这种增长速度。

表1-4　15种最主要生物的蛋白质序列数目（SWISS-PROT数据库39.16版本，2001年4月）

物　种	序列数
智人	6 742
酿酒酵母	4 845
大肠杆菌	4 661
小家鼠	4 269

（续表）

物 种	序列数
褐鼠	2 809
枯草杆菌	2 229
秀丽线虫	2 163
嗜血流感杆菌	1 746
稷酒裂殖酵母菌	1 654
黑腹果蝇	1 443
嗜热古细菌	1 429
拟南芥	1 240
结核分枝杆菌	1 228
牛	1 202
家鸡	948

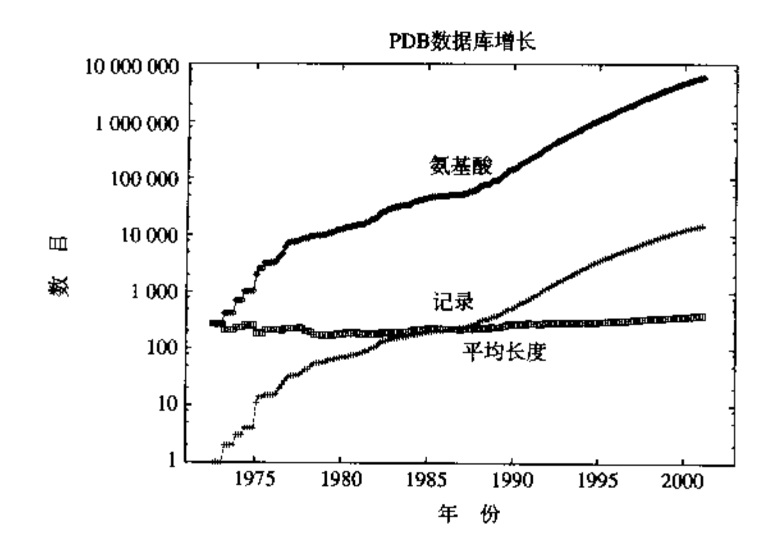


图1-6 1972~2001年间，PDB数据库的指数增长

PDB数据库（2001年4月19日的材料）大约有14 910条记录，共6 033 000个氨基酸（每条记录平均长度为405个氨基酸）。

1.3.3 蛋白质功能

蛋白质的许多功能主要由一些局部序列特征决定，并不依赖于完整的三维结构，这些完整的三维结构部分地由长程相互作用决定。[149] 从全面功能预测的角

度来看，这些特征可以为预测特定蛋白质的准确功能提供某些重要提示。它们在确定蛋白质区室化（compartmentalization）问题的否定结论（例如某个蛋白质是非免疫性蛋白或非核内蛋白）时，也具有重要的作用。

后基因组时代生物信息学领域的一个主要任务是研究基因如何相互配合而发挥功能作用，研究方法是或者利用基因芯片对大量基因的活性进行同时测量，或者分析细胞内蛋白质的互补作用。[408,360,413]很多蛋白质的功能可能很难通过实验手段确定，因为其功能可能与特定生物所生存的自然环境密切相关。酿酒酵母并不是为烤面包而进化来的，而是为了适应其在农作物（诸如葡萄和无花果）体内生活的需要。[215]在基因组中存在的许多基因是在特殊环境中生存的需要，而这些基因在实验室创造的人工环境中可能毫无用处。在许多情况下，我们甚至无法模仿生物的自然寄主及其与其他众多微生物的相互作用，从而无法通过实验确定基因或其产物的确切功能。

为了阐明某些被称为“孤儿蛋白质”（orphan protein）的特殊蛋白质的功能，惟一有效的方法恐怕就是计算分析和预测，这种方法可以得到关于蛋白质功能的很有价值的间接证据。许多蛋白质的特性可以通过序列推断得到。一些序列特征可能和共翻译以及折叠后修饰有关；其他特征可能与结构区域有关，它们可以提供关于特定三维拓扑的证据。沿着这个方向，预测方法可以提供关于蛋白质功能的初步提示，并在以后的实验中加以检验。[288]

近几年来，人们提出了一些并不直接依靠序列相似性的预测方法。[380,162,271,378]一种非常成功的方法就是利用通过DNA阵列[425]和芯片技术得到的基因表达数据（参见第12章）。基因在不同时间点或者组织类型中具有不同的表达模式，我们可以将具有相同表达模式的基因进行聚类。这样，我们可以根据这个类别中最常出现的基因功能（如果这个类别中包含已知功能类型的基因），推断该类别其他未知功能的基因可能具有相同的功能。通过这种方式，功能信息可以在序列相似性很小或者完全不相似的基因之间传递。然而，在很多情况下共调控基因（coregulated gene）的功能差别非常大，因此这种方法不能单独使用。这种方法的另一个缺陷就是，随着DNA阵列的规模越来越大——足以包括一个哺乳动物的全基因组序列，进行聚类时会出现越来越多的具有显著增减调节的基因类别，在这些类别中没有一个基因已知功能信号。

另一种方法则是基于功能域融合模式的，称为“Rosetta stone”方法。[379,167]其基本想法是，如果一个生物体中的两个蛋白质在另外一个生物中是作为一个多功能域蛋白质的不同单元存在，就表明即使这两个蛋白质在序列上没有直接的联系，它们也很有可能执行相同的功能。

可以用来联系具有相似功能蛋白质的第三个工具则是系统进化谱（phylogenetic profile）。[423]在系统进化谱中，每个蛋白质可以用包含其同源蛋白质的生物体来表示。如果两个蛋白质具有相同（或非常相似）的系统进化谱，就表明它们通常可以一起观察到——一个生物体在其基因组中要么同时编码两个蛋白质，要么都不编码。这种现象一个可能的解释就是这两个蛋白质执行相似的功能。随着基因组数据不断增加，系统进化谱有望发挥更大作用。这种方法已经成功应用于酵母基因组，但在我们完成更多的多细胞生物的测序工作以前，这种方法对于人类蛋白质功能预测的作用非常有限。

1.3.4 蛋白质功能和基因本体

在基因组规模上确定蛋白质的功能，需要我们利用一些正确定义的类别、关键词和层次关系，对蛋白质的功能进行系统化的描述。基因本体（gene ontology）本质上就是对分子生物学的相关概念及其相互关系给出规范化的说明。如果科学文献和数据库中的信息要以一种最有用的方式共享，就必须使用一些标准的语法和语义，这就是建立本体的目的。在实践中，这意味着，诸如功能类别和分类学在设计时必须覆盖广泛的生物种类（如果不是全部生物的话），而且这个系统必须能够包含以后的新发现。

这个领域的一个主要进展[21,22]是基因本体论坛（Gene Ontology Consortium）的创建。这个论坛的成员来自于不同的研究领域，所研究的模式生物包括果蝇（FlyBase）、发芽酵母（Saccharomyces基因组数据库），小鼠（小鼠基因组和基因表达数据库），芸苔（Arabidopsis Information Resource）和线虫（WormBase）。基因本体论坛的目标是创建一个动态可控的描述词表。这个描述词表可以反映下述三个组织原则和功能特征：（1）分子功能，（2）生物过程和（3）细胞组分。一个蛋白质能够表现出一种或多种分子功能，参与一种或多种生物过程，并与一个或多个细胞成分有关。

分子功能阐述了基因产物个体所执行的任务，例如转录因子和DNA螺旋酶。生物过程则叙述了更加广泛的生物学目的，例如其中伴随着一些分子功能有序组合的有丝分裂或嘌呤的新陈代谢等。细胞组分包括亚细胞结构、位置和大分子复合体，例如细胞核、端粒和复制起点识别复合体。

我们可以通过许多方式来创建本体，有些方法主要考虑分子复合体或免疫系统，这方面的工作包括RiboWeb本体[123]和ImMunoGenetics本体[213]。另外一个重要的工作是EcoCyc本体，[307,308]用于描述大肠杆菌（*E.coli*）的基因组和生化机制的数据库。这个数据库描述了一系列生物（主要是微生物）的生化通路、

反应和酶。EcoCyc给出了大肠杆菌所有代谢酶的详细描述，包括它的协同因子（cofactor）、激活剂（activator）、抑制剂（inhibitor）和亚基结构（subunit structure）。数据库中还列出了已知的编码某个酶的亚基的基因，以及基因在大肠杆菌染色体图谱上的位置。

1.4 生物序列的信息量

了解信息的概念及其定量化，是理解分子生物学中的机器学习方法的基本原则的最基本条件（附录B是信息论的基本概念，参考文献［577］则是关于信息论的综述）。数据驱动的预测方法应该能从独立样本中提取基本特征，并且摒弃样本中存在的多余信息。这些方法必须能将正样本与负样本区分开。由于基因组中存在大量非功能位点和区域，因此负样本的数量大大超过正样本是很普遍的，机器学习方法必须在这种情况下也能进行正确区分。这种区分问题显然与细胞环境中的分子识别[363,544,474]密切相关：存在大量相似的功能位点时，生物大分子如何从中发现能与其进行相互作用的位点？

机器学习方法在剔除和压缩冗余的序列信息方面有很好的效果。一个规模适宜的神经网络可以利用其可调参数存储许多数据项的普遍特征，而不是单独的序列模式的个别特征。神经网络训练过程中隐含的编码原则在某种意义上将序列进行叠加，从而将输入序列空间的复杂拓扑变换为一种较简单的表示。在这种表示方法下，相关的结构和功能类别最后能聚集在一起，而原先它们在序列空间上通常是分散的。

例如，长度为13的所有氨基酸片段，其中心残基处于螺旋构象中，这些片段在序列空间中是极其分散的。其他类型的蛋白质二级结构，如折叠和转角情况也是如此。在这个序列空间中，存在20^{13}种可能的序列片段（不考虑第21个氨基酸：硒代甲硫氨酸）。人们发现不同的结构类别一般并不位于序列空间中严格分开的区域里；[297,244]相反地，人们可以在倾向于采用螺旋构象的片段所组成的序列区域中发现许多孤立的折叠构象序列，反之亦然。机器学习方法由于其处理非线性的能力，可以发现序列空间中更加复杂的关系（这些序列空间在功能上并未分离），从而在这方面的研究中有用武之地。

一些序列片段甚至可能实现螺旋和折叠构象，具体状态则依赖于先前和其他生物大分子及环境的相互作用。特别需要指出，朊病毒蛋白可能就是这种情况，最近发现，朊病毒蛋白与疯牛病以及人类的克—雅氏综合征（Creutzfeldt-Jakob Syndrome）有关。在朊病毒蛋白中，相同的序列会采用不同的稳定构象：这种构

象由一束螺旋组成，而致病的“坏”构象则由螺旋和折叠混合而成。采用不良构象的朊病毒蛋白具有自催化效应，这可能是将正常构象的朊病毒蛋白转化为不良构象的原因。[266,267,444] 从效果上看，这种蛋白质可以视为可遗传的结构信息的携带者。为了将这种病原体与传统的遗传物质相区分，人们引入“朊病毒”（prion）这个概念以强调其与蛋白质的相似性及其传染性本质。斯坦利·B·普鲁赛纳（Stanley B.Prusiner）由于他在朊病毒方面的研究而获得1997年诺贝尔生理和医学奖。蛋白质可以独自传播传染病的观点让科学界极度震惊，人们对于这些蛋白质功能的内在机制仍有很大的争论。

任何利用局部序列信息的预测方法，都不可能解决类似朊病毒蛋白的构象冲突（conformational conflict）的问题。然而，这些方法可以指出：与卷曲构象相比，某序列片段是否具有更高的实现螺旋和折叠构象的潜能。人们利用一种序列分析中非常成功的机器学习方法——罗斯特（Rost）和桑德（Sander）的PHD方法——分析朊病毒蛋白序列时发现情况正是如此。[266,267] 我们将在第6章中重新考虑这个问题，并讨论其他预测蛋白质二级结构的方法。

另一个与冗余有关的问题是，在确定蛋白质三级结构时不同氨基酸的相对重要性，[347] 即蛋白质氨基酸序列的哪一部分能够完全决定其结构？人们设立了Paracelsus挑战赛，[450 291 449] 鼓励关于（与蛋白质稳定性相对的）序列特异性作用的研究。比赛的内容是将蛋白质在保持原有序列50%氨基酸的前提下，转化折叠成新的蛋白质。最近，一种原来主要由β折叠构成的蛋白质通过这种方法被改变成一种接近自然的稳定的四螺旋束结构（four-helix bundle）。[143] 这些研究表明残基以一种高度非线性的方式决定折叠类型。鉴别出完成特定折叠类型所需要的最基本条件，不仅对于设计预测方法十分重要，而且是解决蛋白质折叠问题的明显进步。[143]

从20世纪50年代末以来，对于生物序列冗余和信息量的分析受到语言学的强烈影响。分子生物学诞生于科学方法论受语言哲学影响的年代。[326] 分子生物学中许多有影响的观点来源于自然语言的哲学和数学处理方法，其原因是这些方法在“自然”生物序列分析中部分地“再利用”，如今这种方法仍适用于分子生物学（参见第11章）。遗传信息的数字化本质以及生物序列可以通过一些连续的步骤翻译成另外一种表示法这个事实，也是建立这两个学科间的联系和类比的重要原因。

人们在破解遗传密码的年代里，翻译遗传密码的研究同样受到语言学的影响。在20世纪60年代，人们将20个氨基酸和翻译的终止信号对应于64个三联体密码子，当时人们认为编码方法最基本的特征是其纠错能力。在那个时代，从宇宙飞船发回的信息中恢复原始消息是编码和信息论领域的一个关键课题。香农（Shannon）

的信息论方法，能够利用冗余性进行编码而在噪声信道上无差错地传输数据，这种方法在当时受到重视。遗传密码的块状结构，确保了密码子—反密码子识别中最经常出现的错误对于翻译产生的影响达到最小程度：所产生的氨基酸不是与原来相同，就是至少具有某些相同的物理化学特性，特别是疏水性（hydrophobicity）。遗传密码其他无错纠正特性的重要性也许被我们低估了。我们将在第6章看到，对于利用核苷酸三联体和氨基酸之间的对应训练神经网络，使用标准密码训练出的神经网络较为简单；而使用其他一些差错纠正密码得到的神经网络则要复杂得多。人们提出这些差错纠正密码，是希望它们能够作为进化过程所产生的编码规则的替代。[524]

生物序列所包含的信息量与它们的可压缩性有关。直观上看，重复量大的简单序列可以用较短的语言描述，而从不自我重复的复杂随机序列所需要的描述语言则要长得多。数据压缩算法在计算机中广泛应用，用于提高磁盘、CD-ROM和磁带的容量。传统的文本压缩策略可以保证在不损失信息的情况下重构原始数据。文本压缩算法可以使用一种冗余较少的表示法给出简短的描述——通常称为代号；我们还可以进行相反的工作：解释代号并还原成未压缩的消息。[447]分子生物学的文献就充满了这样的代号，缩短了这种特殊类型文本的长度。例如：DNA作为脱氧核糖核酸的缩写，帮助压缩了本书的文字数量。[577]

对于一些文本序列，例如计算机程序的源代码，缺少一个符号就可能极大改变原文的意思。但对于其他类型的数据，即使我们不能完全重构原始信息，压缩表示还是很有用的。一个普通的例子就是声音数据。通过电话线传输声音数据时，由于并不严格要求复制所有原始数据，不太精确的解压缩是可以接受的。对于无损压缩，编码后的数据实际上是计算原始数据的一种程序。在后面的章节中，我们对于与机器学习相联系的压缩算法的直接和间接使用，都有详细描述。

在1.2节中，我们叙述了一种用于分析大容量基因组冗余性的实验手段。对于大容量基因组，如果仅仅是对每个基因增加相应比例的拷贝，DNA复性实验的动力学过程会比实验观察到的快得多。由此可以推断，大容量基因组中外显子之外的DNA序列不太可能编码蛋白质，[116]所以对于序列数据的算法压缩不再是一件简单的工作。

对于生物序列中重复片段的统计学特性的研究，尤其是它与基因组进化的关系，可以为我们提供很多信息。这样的分析可以为一些事件提供证据，这些事件比随机产生的单点突变的固定和合并事件更加复杂。相互关联的基因组组合（包括相同物种的个体间作用和物种之间的遗传信息的横向传输）代表了基因组间的相互通讯，这使得对进化通路进行的分析变得十分困难。

大自然产生了一些无用和泛滥的基因组组合，形成不育的生物个体，这些生物体无法对基因库的进化再做贡献。众所周知，骡子是马和驴杂交生成的不能生育的动物。一个不常见的例子是狮虎（liger），这是雄狮与雌虎交配后产生的后代，相应地也存在虎狮（tigron）。与它们的双亲不同的是，这些杂交动物神经紧张、心神不宁，而外表上看它们结合了狮和虎的大部分有代表性的特征。还不清楚能否在自然界发现野生的狮虎，它们的可能的双亲大多居住在不同的大陆①。但在洛杉矶野生动物园中，一些私人拥有者曾经饲养了一些狮虎，不过他们现在不太可能再拥有这些动物。图1-7是这种令人迷惑的动物的照片。

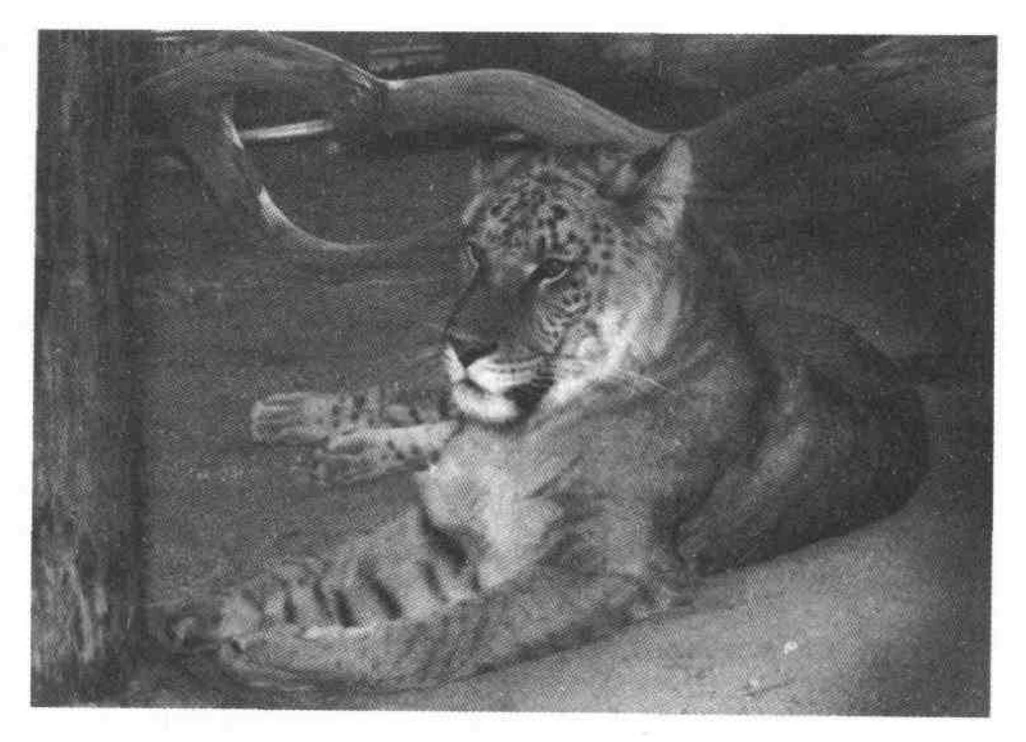


图1-7 雄狮和雌虎杂交的后代——狮虎的照片

洛杉矶野生动物园（Beverly Setlowe）提供。

从描述长度的角度看，生物序列中的重复片段无疑是压缩算法的绝好目标。甚至对于没有重复片段的、自然出现的序列，统计上的偏倚——如二肽、双核苷酸、三核苷酸等分布特性——使我们可能利用代表词（representative word）和延

① 在亚洲的一些地区，狮和虎的活动区域较为接近，如印度西北部的Gujarat地区。

伸字符集（extended alphabet）改写原有序列，从而得到较短的符号序列。

序列编码后的长度与原先长度的比值，就是压缩比率。压缩比率是对数据整体规则程度的定量刻画：

$$R_C = \frac{S_E}{S_O} \tag{1.3}$$

自然文本与DNA序列的一个重要区别在于重复序列出现的情况不同。在较长的自然文本中，重复序列通常较小而且相互近似；而在DNA序列中，长重复序列通常相隔很远。[447]这使传统的序列压缩策略[56]对于DNA和蛋白质数据不太有效。尽管如此，使用为其他类型数据设计的压缩算法，如UNIX环境下基于Lempel-Ziv算法的compress程序，可以得到很好的压缩效果。[551]由于其阅读框和三联体规则性，编码区一般比随机的非编码区如内含子更易压缩，就不令人感到特别惊讶。[279]通常认为，功能性RNA较其他大多数序列少重复，[326]但由于它们折叠成二级结构的潜力很大，因此具有另一种内在结构，从而减少了随机性或信息量。

隐马氏模型是分析序列中单体连续模式（sequential pattern of monomers）的强有力工具。[154]它可以作为给定语言的任意可能序列的生成模型，生成的每个序列都具有相应的概率。由于模型通常要通过训练才能体现给定序列集的规则，大部分序列最后的出现概率非常接近于0。如果训练过程成功，训练集中的序列（可能还有它们的同源序列）的概率值就较高。如果考虑给定字符集的所有可能序列组成的空间，我们可以认为隐马氏模型是对于这个空间上的分布进行参数化的工具。例如，一种特殊的蛋白质家族——球蛋白可以看成序列空间上的一个点集。利用这些序列的一部分训练一个新模型，可以看成是在序列空间上创造一个分布函数，这个分布函数在该点集上的取值高于其他区域。

1.4.1 信息和信息约简

通常从计算的角度看，分类和预测算法意味着减少信息量。算法的输人是含有大量信息的序列数据，输出则可能是一个简单的数。在最简单的情况下，输出甚至只是代表在两类中进行选择的“是”和“否”。对于后者，在两种类别等可能时，输出的信息量达到最大值1比特（bit）。根据氨基酸残基是否位于α螺旋构象进行分类，就是这样一种二分法。在这种情况下，输出的平均信息量会比每个残基1比特小得多，因为自然生成的蛋白质中大约有30%的氨基酸属于螺旋类别。沿着序列“猜测”构象类别时，我们所需要询问的“是/否”问题数的均值小于1。

这些算法的缩减特性意味着它们是不可逆的，我们不能通过反向执行预测程序而返回输入信息。从预测一个氨基酸残基结构类别的神经网络的输出，我们无

法判定输入的究竟是哪个特定的氨基酸，甚至不能断定它与其他氨基酸残基的关系。同样，无论在什么层次上，我们都不可能从隐马氏模型的对数似然度出发重现原有序列。

计算过程一般都会丢弃信息，并以逻辑上不可逆的方式进行。即使简单的数字求和运算也是如此，和数中并不包含各个加数取值的信息。细胞中发生的许多与序列相关的信息处理过程也是如此。遗传密码就提供了一个最显著的例子：64个三联体与20个氨基酸及翻译终止信号之间的对应具有简并性。除了蛋氨酸和色氨酸，其余所有氨基酸都对应多个三联体，这使我们无法从蛋白质的氨基酸序列中得到进行编码的mRNA序列，也不知道究竟是三个终止密码子中的哪一个终止了翻译过程。特定生物个体中三联体的概率分布，即其密码子使用频率决定了实际上翻译过程究竟丢弃了多少信息。

在前面还提到另外一个非常重要的例子，即真核生物通过基因组DNA的转录本mRNA前体生成成熟的mRNA。打断蛋白质编码部分的非编码区（内含子）在细胞核中被剪切，而编码区域重新接合（参见1.1.2和6.5.4节）。但是，看上去很难或不能依靠成熟的mRNA来高精度地定位中断序列所处的接合处，[495,496]显然也不可能从成熟mRNA序列重现内含子序列。剪接接合处保存的局部信息大部分位于内含子中。这点是很有意义的，因为这表明对于组成成熟mRNA序列的外显子，其编码蛋白质的能力没有什么限制。有趣的是，仅仅作为剪接的后果，一些特殊的蛋白质似乎与成熟mRNA序列中的外显子接合处有联系，[256]这意味着剪接后的信使可以“回忆”起内含子的位置。剪接机制在接合处遗留下这样一些特征蛋白质，也许是为了影响转录后体内的后期发生事件——例如mRNA转运、衰变和翻译。

信息约简的更为奇异的例子当属RNA编辑现象[59]和从蛋白质中移去“内含肽”（intein）。[301,257]在RNA编辑过程中，利用基因组中位于其他位置的指导RNA序列对于原始转录本进行后期处理。这个过程中，单个核苷酸或更长的片段被改变，或者直接跳过。很明显我们无法从编辑过的mRNA序列恢复基因的原始RNA拷贝。

人们发现在某些多肽链中也存在剪接现象，内含肽序列片段被移开，随后不同序列的终端接合在一起。在嗜热古细菌的全基因组中，人们惊讶地发现在预测所得的开放阅读框中存在大量的内含肽。逻辑或者物理不可逆的其他例子也大量存在，这显然与大部分生命过程的不可逆的热力学本质有关。

计算分类和预测方法的信息约简本质，使我们更容易了解为什么在对数据进行处理时加入额外的数据通常并不会产生更好的效果。如果所添加的数据并

没有很强且有价值的关联，结果只是使预测算法在输出只有1或2比特的结果时要丢弃更多的信息，从而增加了计算负担。尽管事实上额外的数据包含某些可利用的特征，但所得的结果经常具有较低的信噪比，从而降低了预测性能（参见第6章）。

蛋白质二级结构预测通常在使用13个氨基酸片段数据时会取得更好的效果，而不是长度为23或更高数值的序列片段。这不仅仅是由于输入空间的维数灾难问题：维数越高，固定数目的样本分布在越稀疏的空间。[70] 而是因为实际情况中，尽管序列数据中存在长程相关性，但若我们所拥有的蛋白质三维结构的数据量是一定的，上下游额外10个残基加入的噪声就会超过长程相关所带来的正面效应。

在处理非关联的数据特征时，机器学习方法由于具有内在的鲁棒性（robustness），与其他方法相比具有很大的优势。神经网络的权重因子在训练过程中会消失，除非序列数据的正相关性或负相关性将它们保留下来并加以利用。这意味着使用23个氨基酸并不会成为一场灾难，但是如果要求所需要处理的输入空间的信号与噪声的关系更加和谐，利用23个氨基酸的预测算法效果仍不理想。

信息约简对于我们理解几乎所有类型的预测系统都是一个关键。如上文所述，机器学习算法可以得到序列空间更简单的表示方式，这种表示方式比包含所有细节的原始数据更有用，功能也更强大。

《艾丽斯漫游仙境》的作者，数学家查尔斯·道奇森［Charles Dodgson，笔名为刘易斯·卡罗尔（Lewis Carroll）］在100多年前就讨论过地图与映射关系的实际问题。在“西尔维和布鲁诺加入队伍”这一节故事中，米恩·赫尔（Mein Herr）描述了人们所能想到的最广阔的地图，一个比例尺为1：1的地图。他被问道：“你使用过这个地图吗？”他答道：“它至今还没有被展开。农夫们讨厌这个地图，他们说这幅地图会覆盖所有的土地并遮住阳光！现在我们正用着这片土地，它就是它自身的地图，而且我保证它近乎完美。”

按照米恩·赫尔的观点，我们应该保留现有公共数据库中文本结构松散、平面化的特点，不要试图利用神经网络或者隐马氏模型等工具增强数据库文件的主要特征。

1.4.2 比对和预测：比对何时可靠

为了得到更多对于功能的认识以及关于结构和功能关系的暗示，我们通常会将新序列与许多数据库中的所有序列进行比对。[79] 一个基本问题是：对两条序列进行比对时，序列相似性需要达到什么程度才可以使我们放心推断两者的结构或功能具有相似性？换句话说，如果通过某种比对方法检测到一个序列片段发生

重合，我们能否由此定义一个相似性阈值用于筛选可靠的检测结果吗？在阈值以下，进行比较的两条序列有些相关、有些不相关，因此低于阈值的匹配并不足以得到否定结论。众所周知，在序列相似性很低的情况下，蛋白质的结构也可能非常相似。而在相似性标准这么低的时候，会由于随机性产生其他相互比对的结果，从而与真正相关的序列所产生的比对结果相混淆。

对于这个问题较有意义的回答是，这完全依赖于你要考察的那个特定结构或功能特性。对于不同的目的，充分和必要的相似性阈值是不同的。可靠的结构性推断仅需要一个层次的相似性，而功能性推断对于每一种功能一般都需要一个新阈值。某些功能特征可能和全序列有关，例如一个序列是否属于一类特定的酶。而另一些功能特征则完全依赖局部序列的组成，例如，一条蛋白质序列靠近N端的特定位置是否有信号肽剪切位点。

一般而言，在推断较有把握的范围内，人们更倾向于进行序列比对而不是预测。在最好的情况下，预测方法应该可以扩大可靠推断的区域。许多方法在给出预测结果的同时也给出了置信度，因此我们可以通过对置信度进行评价而实现上述目标。我们将在第5章详细讨论这方面的内容。

桑德（sander）和施奈德（Schneider）首先对蛋白质序列相似性与结构相似性的关系进行算法研究。[462] 在比对长度与重合部分中相同残基所占百分比的关系图中，可以看到两个区域：一个区域完全由结构相似的成对序列组成，另一个则是既有相似组又有不相似组的混合区域。重合部分中大于70%的残基二级结构类别相同，就被定义为结构相似。可以观察到对应于三维空间中两个序列结构比对的均方根偏差的最大值为2.5埃。混合区域反映了二级结构相似性偶然可能超过70%，尤其对于非常短的重合片段，甚至完全无关的一对序列的相似值也可能很高。

两个结构域的边界和由此得到的用百分比衡量的序列相似性阈值，依赖于比对（重合）区域的长度。桑德和施奈德定义了一个与长度有关的阈值函数：重合长度$l<10$时，认为没有相似性；$10<l<80$时，阈值为$290.15l^{-0.562}\%$；$l>80$时，阈值为24.8%。

我们可以利用阈值，考察序列比对方法是否可以得到可靠的推断，或者对于特定的目的我们是否必须使用预测方法。如果新序列的相似性高于阈值，倾向于用比对或同源模建的方法；如果低于阈值，则应该采用更加先进的模式识别技术的预测方法，或者将其和比对方法结合使用。

对于此类分析，所谓“推断的可靠区”当然不是100%可靠，而仅仅作为一种指导思想，例如为验证高通量预测算法而构造测试集。我们知道在许多情况下，一个氨基酸的改变会导致蛋白质变成一个完全不同，甚至可能是不折叠且不发挥

功能的蛋白质。单核苷酸多态性项目的部分目标就是为了鉴别位于编码区的SNP位点，这些位点可能影响蛋白质构象，并由此影响疾病的易感性，并且通过和特定蛋白质发生作用而改变药物疗效。[394]

1.4.3 功能特征的预测

用于解决结构问题的结构序列一致性的阈值，不能直接应用于与功能相关的序列预测问题。如果我们要根据数据库中已通过实验验证的一些序列的信号肽剪切位点，来准确推断一个新序列位点的确切位置，则我们事先完全不知道所需要的相似性程度应该是什么。

在上文中，通过量化空间中的平均距离来定义“结构相似”。在比对算法中，功能相似则意味着具有相似功能的任何两个碱基必须精确匹配，没有任何偏移。如果希望根据一个序列的剪切位点明确定位另一个序列的位点，则两个序列在剪切位点附近必须严格对齐。实际上，能否仅通过比对将完全可靠的推断区域与混合区域正确区分，这依赖于不同类型功能位点的保守程度。

与区域分离原则的定义相结合，成功比对的二元标准可以用于确定阈值函数，该函数能给出功能相似性的最佳辨别效果。[405]确定一个精心选择的阈值有一个通用原则：这种方法应该容易推广到其他类型的序列分析。这类分析包括确定蛋白质序列的糖基化位点、磷酸化位点，叶绿体和线粒体的转运肽（transit peptide）以及多蛋白的剪切位点（cleavage site）。对于核酸序列，这类分析包括mRNA前体的内含子剪接位点（splice site）、核糖体接合位点以及启动子。对于每类分析必须给出对应的阈值。

对于诸如mRNA前体剪接位点和蛋白质糖基化位点的这类分析，必须考虑到一个序列上存在多个位点。解决这种问题的一种方法是将每个序列分割为多个子序列，每个子序列有一个可能的位点，然后对于子序列利用以上算法进行预测。另一种方法是利用每个比对过程中所对齐的位点所占的比例，作为衡量功能相似性的一个指标，这与桑德和施奈德所使用的结构相似性（比对中具有相同二级结构的氨基酸所占的百分比）有些类似。对于这种情况，在具体计算功能相似性阈值之前，必须先给出功能相似性的阈值的定义——对应于桑德和施奈德使用的70%结构相似阈值。

1.4.4 全局和局部比对以及替换矩阵的熵

实际上，利用对序列的整个结构都普遍适用的规范或惟一的评判标准，并无法评判两个序列的两两比对是否最优。比对算法（alignment algorithms）所得出

的匹配结果完全依赖于定义对应单体相似性的量化参数、空位和缺失的罚分，而算法的设计目标是用于优化全局还是局部得分对于结果的影响最为显著。

有些关于生物相关的问题必须对两个序列进行全局比较，也可能不考虑序列末端的差异；而有些问题，如果不从子序列的角度对具有相似序列结构的片段或者位点进行定位，就变得毫无意义。

传统的比对算法是基于动态规划的，如最优全局比对的Needleman–Wunsch算法[401,481]和最优局部比对的Smith–Waterman算法[492]（参见第4章）。如果我们要对两个序列任何可能的比对所对应的分值进行穷举评价，其计算量是以组合方式增长的。动态规划是一种可以用于控制计算量组合爆炸（combinatorial explosion）的计算方法，然而动态规划的计算量依然很大。为了进一步减少发现显著比对所需的数据资源，人们设计了一系列的启发式算法。[417,419]其他一些速度很快而且很可靠的启发式策略并不是建立在动态规划基础上，而是交互式地将较短的子序列延伸为更长的匹配。[13,14]读者可以在其他文献中[550,428]找到关于传统比对策略的详细叙述。这里，我们将集中考虑与精心设计数据集有关的一些重要问题。

在实际应用中，替换矩阵（substitution matrix）的选择在很大程度上影响局部比对策略的“局部”程度。如果匹配的分值比错配的罚分高得多，即使局部比对策略也会得到较长的比对结果。如果错配的分值将匹配得分完全抵消，一般会得到短而且紧凑的重合结果。

替换矩阵给出了一组分值s_{ij}，分别对应于将氨基酸j用氨基酸i替换。一些矩阵是通过一种简化的蛋白质模型产生的。这个模型包括氨基酸出现频率p_i和两两替换频率q_{ij}这两个参数，这些参数可以通过已有的自然生成的蛋白质之间的比对得到。稀有氨基酸之间的匹配得分可能比普通氨基酸之间的匹配得分大，而两个可以互换的氨基酸之间的错配，其得分也要比两个功能上毫无关联的氨基酸之间的错配得分大。分值为非负数的错配代表某种相似性或者保守替换。其他类型的替换则基于氨基酸之间的关系。这种关系主要体现在氨基酸的遗传密码、物理化学特性，或者是氨基酸本身在比对中是否一致等方面。

所有这些不同的替换矩阵都可以建立在替换矩阵熵概念的基础上，并相互进行比较。阿特休尔（Altschul）指出，[8]任何氨基酸替换矩阵，都可以直接或间接地看做归一化目标频率的对数值所组成的矩阵。这是由于替换矩阵可以写成

$$s_{ij} = \frac{1}{\lambda}\left(\ln \frac{q_{ij}}{p_i p_j}\right) \tag{1.4}$$

其中λ是缩放因子。改变λ会改变绝对分值，但不改变不同局部比对的相对分值，因此不会影响比对结果。[405]

最简单的计分矩阵是恒等矩阵，该矩阵中所有对角元素取相同的正值（匹配分值s），而所有的非对角元素取相同的负值（错配分值$\bar{s}$）。尼尔森（Nielsen）曾经讨论过这种情况。[405]同一矩阵可能来自于可能的氨基酸替换的最简单模型，这个模型中，20个氨基酸等概率出现，并且非对角元素代表的替换出现的概率也相同：

$$p_i = \frac{1}{20} \qquad \text{对于所有} i$$

$$q_{ij} = \begin{cases} q & \text{对于} i=j \\ \bar{q} & \text{对于} i \neq j \end{cases} \tag{1.5}$$

换句话说，如果一个氨基酸发生突变，它变为其余19个氨基酸的概率都是$\bar{q}$。

根据匹配分值和错配分值比率$s/\bar{s}$的不同，存在一系列不同的恒等矩阵。如果$s=-\bar{s}$，局部比对要得到正分值，所包含的匹配要比错配多得多，由此产生短而强的比对结果；如果$s \gg -\bar{s}$，一个匹配就可以补偿多个错配，产生长而弱的比对。在不考虑空位的恒等矩阵局部比对中，两条序列匹配氨基酸所占的比例p有最小值，即

$$p > \frac{-\bar{s}}{s-\bar{s}} \tag{1.6}$$

我们定义突变性$r=\bar{q}/q$，即序列特定位置上的氨基酸改变成随机氨基酸（包括原来的氨基酸）的可能性。r=0对应于没有发生改变，而r=1则对应于进化距离无限大。

由于所有q_{ij}的总和必须为1，利用关系式$20q+380\bar{q}=1$可以计算目标概率

$$q = \frac{1}{20+380r} \quad \text{和} \quad \bar{q} = \frac{r}{20+380r} \tag{1.7}$$

并通过（1.4）计算s_{ij}的取值。由于分值比率$s/\bar{s}$与λ无关而是r的函数，我们可以通过分值比率计算r值。

对于氨基酸矩阵，阿特休尔如下定义相对熵：

$$\mathcal{H} = \sum_{i,j} q_{ij} s_{ij} \tag{1.8}$$

其中s_{ij}的取值都经过归一化，因此$\lambda=\ln 2$［对应地，（1.4）中使用以2为底的对数］。矩阵的相对熵可以用来解释比对中每个位置所携带的信息量。（读者可以

从附录B中了解熵和相对熵等所有信息论方面的概念。）

计算替换矩阵时所假设的进化距离越短，对应的$\mathcal{H}$就越大。如果进化距离为0（r=0），则错配罚分$\bar{s}$为无穷大，这代表完全不允许空位；相对熵也等于氨基酸分布的熵：$\mathcal{H}=-\sum_i p_i \log_2 p_i$。对于恒等矩阵模型，$\mathcal{H}=\log_2 20 \approx 4.32$（比特），局部比对问题就简化为寻找两个序列中最长的相同子序列。相反，如果进化距离趋向于无穷大（r≌1），q_{ij}取值的所有差异消失而且$\mathcal{H}$趋于0。

1.4.5 保守序列和序列标识

研究分子结合位点的特异性时，一种通常的方法是首先从比对中生成保守序列（consensus seqnence），并选择最常出现的核苷酸或者氨基酸作为特定位置的代表。[474]这样的过程丢弃了许多信息，而且如果将所得到的结果当做识别蛋白因子或核酸的分子特异性的可靠评价，则会严重误导进一步的研究。一种较好的替代方法是同时观察所有位置上所有核苷酸（或氨基酸）出现的频率。

施奈德及其合作者在序列每个位置的香农信息量的基础上，发展了一种图形可视化技术——序列标识方法（sequence logo approach）。[473]这种思想强调了单体出现频率对均匀分布的偏离。在均匀分布中，所有单体出现的概率为相同值p。对于核苷酸序列比对，p=0.25；对于氨基酸序列比对，p=0.05。

大多数功能位点附近的频率分布与均匀分布都有显著的偏离。对于特定位置i，实际观察到的单体出现频率与随机状况的偏离可以通过以下公式进行计算：

$$D(i) = \log_2|A| + \sum_{k=1}^{|A|} p_k(i)\log_2 p_k(i) \tag{1.9}$$

其中$|A|$表示字符集的字符个数，一般为4或20。由于计算公式中使用以2为底的对数，$D(i)$的单位通常用“每个单体多少比特”来衡量。对于氨基酸比对，仅当特定位置上只有一个完全保守的氨基酸时，$D(i)$取最大值，为$\log_2 20 \approx 4.32$。同样，对于核苷酸序列比对，偏差最大可以达到2比特。

使用序列标识可视化技术时，可以利用一列符号显示一条保守序列的细节。列的总高度等于$D(i)$值，而每个单体符号k的高度与它在这个位置上的概率$p_k(i)$成正比。利用不同的颜色描绘单体可以表示其物理化学性质，如带电量、疏水性，也可以表示核苷酸相互作用特性，如氢键势能的强弱。与覆盖比对区域的权重矩阵中的数字相比，标识技术的功能强大且很容易使用。如果对位点周围区域的D进行叠加，可以得到特定类型位点（例如结合位点）的累积信息。D可以代表结合位点的结合能的强弱，并用来比较在全基因组和蛋白质序列中寻找真位点所需

的信息。[474]利用信息论中表述序列保守程度的公式，我们可以将蛋白质如何从大量假位点中发现所需的结合位点这个问题以一种定量的方式解决。[474,472]

图1–8和1–9给出两个比对的例子，它们利用标识技术显示单体出现的频率。第一个例子来自大肠杆菌中翻译起始位点的比对结果。在真核基因组的细胞核中，起始三联体——起始密码子——非常保守，几乎总是AUG，代表蛋氨酸。对于原核生物，其他几个起始三联体出现的频率也较大，而标识方法可以显示这三个密码子位置的保守程度究竟多大。[422]由于相同的大肠杆菌核糖体复合体需要识别所有的翻译起始位点，因此标识方法可以指出核糖体复合体与三联体序列互相作用的特异性。紧靠起始密码子5′端的保守的夏因—达尔加诺（Shine–Dalgarno）序列可以通过碱基配对，将mRNA定位在核糖体的正确位置。

如果进行计算时，仅仅使用具有相同信号的序列得到的标识信息量很大。但是这种方法也可以用于辨别具有差异的模式，这些模式属于数据的不同部分。对于极端嗜热古细菌（*Sulfolobus solfataricus*）来说，翻译起始模式还依赖于基因处于操纵子（operon）内部还是操纵子的起始端，或者仅仅是一个孤立的基因。在最近的研究中，[523]在操纵子内部基因的上游发现了一条夏因—达尔加诺序列，但是这条序列并不与操纵子的第一个基因或孤立的基因相对应。这表明这种生物使用两种不同的翻译起始机制。

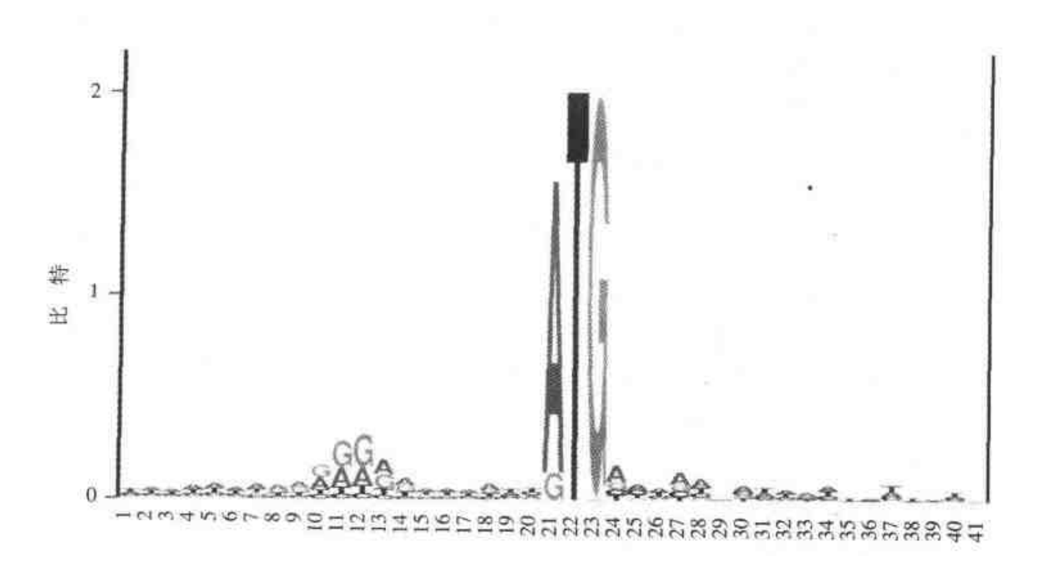


图1–8 大肠杆菌中的翻译起始三联体比对的标识

图中的翻译起始位点在21位点，编码蛋氨酸的起始三联体ATG在标识中异常显著且居主导地位。资料来自参考文献［422］。

图1-9是一些哺乳动物氨基酸序列片断的标识，这些片断利用α螺旋的起始位置加以对齐。[99] 标识覆盖了过渡区域：左边最经常出现卷曲和转角构象，而右边则可以在这列符号的上方发现一些在α螺旋中频繁出现的氨基酸。有一个现象很有意思，在N端即α螺旋的帽状结构部位，氨基酸的分布比螺旋自身中的分布更具有偏向性。[435] 而C端螺旋的标识显示出另外一段的加帽作用（capping）。帽状结构部分的残基很有可能也是这种类型二级结构的一个集成部分，通过其侧链的氢键稳定了螺旋中的偶极。[435] β折叠中也具有一个相似的定界区域，称为β断点，它代表这种结构的终止。[133]

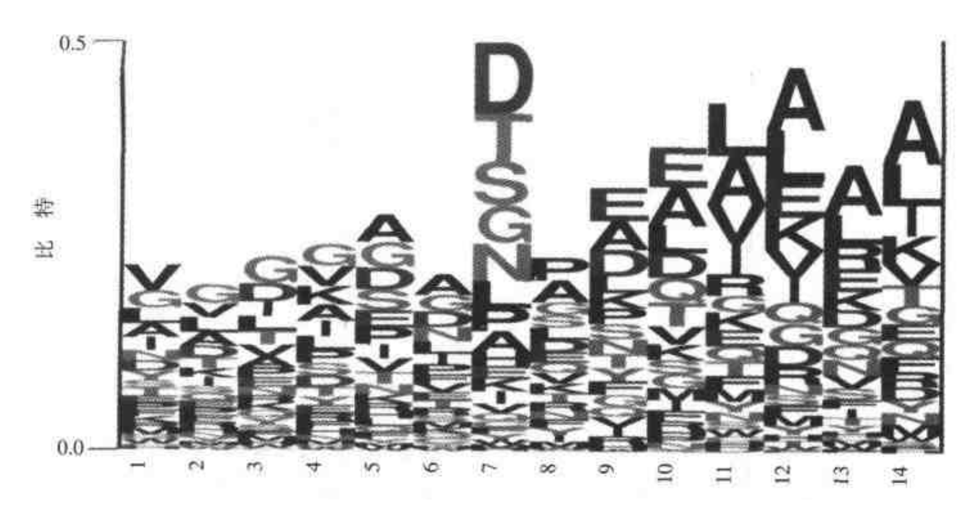


图1-9 α 螺旋N端比对的标识

所用的数据是已知三维结构的哺乳动物蛋白质的非冗余数据组。[99] α螺旋的起始位置在标识中的位点7。二级结构由卡布希（Kabsch）和桑德算法确定。[297] 该区域内构成上的最大偏倚位于螺旋起始点之前。

序列标识对于迅速考察功能位点的上下游或者某些区域的功能特性非常有用，而且它们还能显示在多大程度上存在某种序列信号。如果对很多糖基化位点进行比对并观察其标识，就可以立刻揭示其氨基酸组成的差异程度。这样的分析不仅可以用来构造预测方法的结构，而且可以用来考察哪些样本可以被准确预测。如果糖基化的丝氨酸和苏氨酸上下游具有共同的特性，在设计预测算法时可以将两者结合考虑。[235] 如果两者的上下游序列的特性有显著差异，则必须针对两种残基类型分别设计不同的方法。在细胞环境中，这样一种差别也提示了将糖类转移到这两个残基上的酶并不相同。

使用单体的序列标识方法，实际上是独立对待位点上下游的每个位置的。从

标识中无法得到不同位置的关联情况，也无法了解不同单体同时出现的频率是否超过了利用单独位置的统计量所预测的标准。不过，这种可视化技术很容易处理双核苷酸或者双肽出现的频率，并以符号组合的方式显示位置之间的两两关联。（1.9）中字符集的大小$|A|$要做相应变化，而公式的形式仍然不变。

图1–10是一个双核苷酸的标识，序列样本来自于植物阿布属拟南芥（*Arabidopsis thaliana*）内含子的剪接供体（donor）位点。在标识中部，我们可以观察到众所周知的位于剪接接合处的保守双核苷酸GT和GC（几乎看不到）。标识还显示出，在内含子区域第三个双核苷酸位置，GT出现的频率远远超过预想的值。

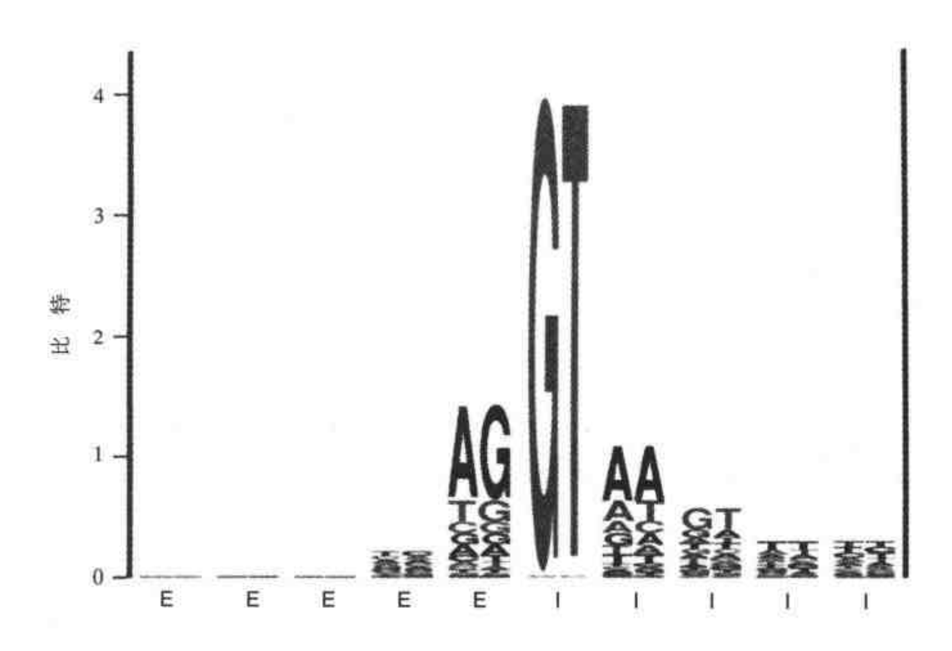


图1–10　双子叶植物阿布属拟南芥的内含子剪接供体位点的标识

该标识基于外显子/内含子过渡区域中不重叠的双核苷酸出现频率，在（1.9）中取$|A|=16$使用标准的香农信息量加以计算。数据集来自于从GenBank中提取的非冗余数据组。[327]

公式（1.9）的另外一种略有差别的形式是基于相对熵或Kullback-Leibler不对称散度（asymmetric divergence measure）的，[342,341]如下式所示：

$$\mathcal{H}(i)=\mathcal{H}\left(P(i),Q(i)\right)=\sum_{k=1}^{|A|}p_k(i)\log\frac{p_k(i)}{q_k(i)} \tag{1.10}$$

这个标识公式量化了观察到的概率分布$P(i)$与参考概率分布$Q(i)$之间的区别。

Q不一定依赖于比对中的位置i。在显示相对熵时，每个字符的高度作为出现频率的替代指标，还可以通过相应位置的背景标定频率得到。[219]

为了让标识成为特异性的可靠描述，进入比对的数据必须不存在冗余，这是一个基本条件。如果多个序列都包含了某个特定位点，就会对概率分布产生影响。

简单的标识可视化技术以及与其相对应的权重矩阵方法，都是对矩阵中每个位置的独立分析。我们将会在第6章中阐述神经网络如何进一步拓展这种分析方法。通过计算“正”位点比对结果中单体的出现频率与参考分布中频率的比值，权重矩阵对每个位置赋予相应的权重值。给定一个序列，如果将每个位置上单体所对应的权重的对数值求和，可以得到一个分值。我们还可以调整选取一个阈值以实现对于真位点的最佳识别。所谓最佳的标准，是根据预测方法的敏感度或特异度加以考察的。

神经网络具有对序列数据进行非线性处理的能力，因此在计算时可以考虑不同位置之间的相互关联。“非线性”本质上意味着：在一个类别与两个特征中的一个相关，而不是与两者同时相关的情况下，网络有可能进行正确的预测。而线性方法则不能正确处理这种两个特征的情况。

对于更加复杂的情况，一个特定类型的位点可能要用许多特征加以表示，相互之间的关联也具有更加复杂的模式。通过特征定义正确处理这种情况的能力，使神经网络算法成为序列数据分析领域的重要工具。

糖基化位点就是这样一个例子，其中带有正电荷和负电荷的氨基酸都可能是糖基化位点并发挥功能，然而这两种类型的氨基酸不能同时出现。传统的单体权重矩阵不能处理这种普通情形。当然，对于某些预测问题，可以利用双肽或输入特征更加复杂的权重矩阵来避开这个困难。另外一种处理策略是将所有的正样本分成两类或更多类，每一类都用相应的权重矩阵代表。这样一种方法上的转变在某些情况下可以有效地将一个非线性问题转化为线性问题。

一般而言，线性方法的缺陷在于不能减少无用的证据。对于线性方法，甚至在生物机制每次只能接受两种证据中一种的情况下，也必须将两种类型的证据结合并相加而得到较高的分值。非线性方法则可以避免这种情况，只要在许多特征的组合证据超过某个标准时，简单地降低分值即可解决这个问题。

对于许多问题，将输入数据的表示方法进行某种良好的变换，实际上是序列空间拓扑结构转换工作的一部分。这种转换将序列空间转为联系更紧密的空间，从而使许多孤立的数据集可以根据它们所属的功能类别相互融合。由于序列中的相关性和特征经常在很大程度上是未知的（至少在我们开始预测分析时），因此

神经网络的非线性处理能力使其在执行许多不同任务的初期具有很大的优势。

多年来，选择何种分析方法在人工智能领域一直是一件非常武断和个人化的事情。在生物序列数据研究领域，显然如果我们事先知道所要寻找的特征，那么多种不同方法的性能大致相当。在分析为特定任务而训练的神经网络的权重（参见第6章）时，如果发现某些特定的序列特征会提高（或降低）神经网络的预测结果的正（或负）预测性能，经常就可以从中构造出同样具有很高区分能力的判别规则。许多人在研究中体会到：若数据相对规范，就能够迅速构造近似最优方法，这种情况下机器学习方法是很有效的，而要从杂乱无章的数据直接归纳很有用的规则就要艰难得多。

1.5 生物分子功能和结构预测

冯·海因（von Heijne）在他早期关于序列分析的书中提出这样一个问题："当你得到序列信息之后能做些什么？"[540]本书所描述的方法和应用就是为了回答这个问题。适于应用机器学习方法进行处理的问题将在以后的章节详细阐述。这里我们将列出一些在概率理论体系下分析DNA、RNA和蛋白质序列数据时需要解决的计算问题，并加以评述。对于某些情况，我们会用实验测定的生化特性表示特定序列，而不是有限字符集中的符号。

1.5.1 序列分析

无论分析的是细胞中的DNA还是RNA，大多数情况下我们将使用单链序列。一个例外是分析DNA的结构元件，例如可弯曲性（bendability）或内在弯曲势（intrinsic bending potential），此种分析必须建立在双螺旋的真正的双链阐述基础上。

真核生物mRNA前体的内含子剪接位点和分支点。中断RNA和蛋白质编码基因的间插序列可以用接合处的局部特征刻画，但是不能由这些特征完全决定。预测蛋白质编码基因中的内含子是极具挑战性的计算问题。对于一些生物（如酿酒酵母），核基因组的内含子很少而且其剪接位点很保守。而对于包括人在内的其他许多真核生物，准确界定编码区和非编码区，并由此从基因组DNA序列中预测成熟的mRNA序列是一个重要的问题。酵母中的内含子主要位于编码核糖体蛋白质的基因中。许多生物的基因，会根据组织形态和发育阶段的不同，选择不同的剪接方式，这个事实使问题更加复杂。为了解决这个问题，人们已经采用许多不同类型的权重矩阵、神经网络和隐马氏模型进行分析。

原核生物和真核生物的基因发现。机器学习方法已被用于通过计算方法发现基因的几乎所有步骤。这些步骤包括翻译起始点和终止点的确定，定量刻画潜在的阅读框（reading frame），剪接位点的框中断，外显子识别，基因建模以及序列拼接等。通常会将多种不同机器学习方法组合起来应用。

启动子识别——转录起始和终止。转录起始是基因表达的第一步，构成了生物体控制的一个关键点。当RNA聚合酶（催化从DNA模板制造RNA的反应的酶）识别启动子序列并与之相结合，表明转录起始事件发生。这类预测问题的困难之处在于作为聚合酶识别底物的DNA信号相似程度各异，而且在表达水平的调控中还有其他许多调控因子参与。隐马氏模型灵活的匹配能力对于解决这个问题——尤其是对于真核生物——非常理想，但是人们也采用了输入结构经过精心设计的神经网络。

基因表达水平。如果我们已经通过实验得到基因序列与其表达水平之间的关系，就可以利用预测已知启动子信号强度的方法来推测基因表达水平。一种替代方法是根据编码序列来预测基因表达水平，这时人们利用密码子（codon）使用频率；在某些情形下，还可以使用相应的密码子适应指标对序列的统计量进行编码。

DNA弯曲及其可弯曲性的预测。双螺旋的柔性（flexibility）影响或决定了细胞内的许多反应。其中之一就是转录起始，利用序列信息预测转录起始或者螺旋的曲率/可弯曲性，对于理解大部分与DNA有关的现象很有价值。

核小体定位信号。与DNA柔性相关的一个问题是：真核生物的DNA在染色质中被组蛋白八聚体缠绕时，该DNA序列的位置关系如何。由于这种位置信号间隔10.1~10.6bp，或随着双链螺旋的每个完全扭转而出现，因此检测周期性的方法需要对于非整数值敏感——就像隐马氏模型中的弹性匹配能力。

序列聚类及类的拓扑结构。由于序列数据不可避免地存在冗余，所以将序列组成一定类别的聚类技术很重要，与此同时还要估计组间的距离。自组织图形式的神经网络和隐马氏模型在这方面非常有用。与其他聚类技术相比，这两种方法的一个优势在于能很好地处理由几千条序列组成的大数据集。

RNA二级结构预测。当前，对于mRNA、tRNA和rRNA可能的二级结构进行计算和排序最为有力的方法是基于能量最小化原则。这里的能量包括碱基对的两个碱基之间或两个碱基对之间的自由能，以及碱基对的堆积能量。[528,260]由于许多原因，如环之间的相互作用使所需评价的结构数目十分巨大，使得这种方法存在很多困难。与传统的寻找最佳能量构象的最小化方法难以成功相比，神经网络和文法方法在处理这些特征时取得了一些成功。

DNA和RNA的其他功能位点和类别。人们对许多不同类型的功能位点和类别也分别进行预测，其中包括：内含子的分支点，核糖体结合位点，蛋白质—DNA相互作用中的motif（超二级结构模体），[②] 其他调控信号，DNA螺旋类别，限制酶切位点，DNA解链温度，EST序列中的阅读框的断性缺失（reading frame-interrupting deletion），根据系统进化类别对核糖体RNA的分类，以及根据物种区别对于tRNA序列的分类。

蛋白质结构预测。这个领域的研究促进了机器学习方法在序列分析中的应用，钱（Qian）和塞诺斯基（Sejnowski）在关于蛋白质二级结构的预测工作方面做出了显著的贡献。[437] 事实上蛋白质结构的所有方面都已经利用机器学习方法进行过处理。可以精确列出的预测内容包括：二级结构的类别，残基接触的距离限制，折叠类型，二级结构的分级或内容，半胱氨酸间的双硫键，蛋白质家族的隶属关系，螺旋跨膜区及其对应细胞膜的拓扑，膜蛋白类别（跨膜区域的残基数），MHC motif以及氨基酸的水溶性（solvent accessibility）。

蛋白质功能预测。预测所研究的与功能相联系的特征包括：亚细胞定位，分泌性蛋白的信号肽剪切位点，信号肽剪切位点的重新设计（用于优化剪切效率），信号锚（signal anchor）（Ⅱ型膜蛋白的N端部分），与糖类相结合的糖基化位点（糖基化的状态和类型决定了循环的周期，这在现象识别和分选中有重要作用），与转录后修饰有关的磷酸化和其他修饰作用（磷酸化位点的存在表明相应的蛋白质参与了细胞间的信号转导、细胞周期控制，或作为营养和环境压力信号的中介），蛋白质的不同结合位点和激活位点（与酶的活性相关）等。

蛋白质家族分类。要预测蛋白质的家族关联，可以对蛋白质的二肽频率进行全局编码，并将其输入自组织图和前馈神经网络。此外基于局部motif的预测也有助于探测较远的家族关系。

蛋白质降解。在所有生物中，蛋白质都必须降解并且再循环。在具有免疫系统的生物中，降解的特异性对于实现免疫系统功能及正确区分自身和异体分子是非常基本的。激活降解通路有许多不同方法。在许多通路中，蛋白质都要在蛋白酶水解切割前先进行解折叠，由此可以推测，蛋白质的特异性是与序列模式而不是与它的三维结构密切相关。因此，这类问题非常自然地仍要利用机器学习方法解决，而面临的主要困难是实验确认的数据量很有限。

② “motif”一词在文中用于表示蛋白质的超二级结构模体或者一段具有特色功能的生物序列（此处表示超二级结构模体），没有较为恰当的中文对应词，因此在文中不予译出。——译者注

第2章　机器学习的基础：概率理论体系

2.1　简介：贝叶斯建模

机器学习基本上直接来源于一门古老的概率学科：统计模型拟合。和统计模型拟合一样，机器学习的目的是通过建立适当的统计模型，从一个数据集D中找出有用的信息。机器学习的一大优点就是可以灵活构建由大量参数刻画的模型，由机器自动处理数据，使信息提取过程尽可能地实现自动化。计算机学习的灵感来自于生物大脑的学习能力。因此，使用了一个特殊的词汇“学习”来刻画这一统计模型拟合过程。

显然，以下两方面技术的迅速发展促进了机器学习方法的发展：

- 可支持大规模数据库和数据集的感知器和储存器设备
- 可处理更复杂模型的计算能力

正如参考文献［455］中指出的，机器学习方法适用于那些拥有大量数据但相应理论很不完善的领域。这正是计算分子生物学使用机器学习方法的原因。

随着序列数据的迅速增长，与有待发现的生物学知识相比，我们现有的生物学知识非常有限。在生物学以及其他数据信息丰富的学科，特别是计算生物学中，人们必须认识到现有知识尚具有高度的不确定性：许多知识是未知的或原本就是错误的。所以计算生物学家经常要面临归纳和推论问题：利用可处理的数据建立模型，发现或修正未知的或现有的生物学知识。什么是正确的模型类型以及什么样的模型复杂度合适？哪些细节重要，而哪些可以忽略？如何根据已有的知识和有时是有限的数据，比较不同的模型并选出最好的一个？简而言之，我们怎么知道一个模型是好的模型？这些问题在机器学习方法中更显得重要，因为复杂模型

的参数通常是几千个甚至更多，并且序列数据（经常是冗余的）本身存在噪声。

在数据不足的情况下，试图通过设置某些参数，使机器学习中使用的模型能够反映研究对象几乎所有的行为，这是不现实的。而为了避免模型的过拟合，更倾向于采用一些包含较少参数的简单模型。深入了解机器学习理论的人们清楚地知道，许多约束条件隐含在模型结构之中。所以，让模型完全重现研究对象的行为，这是极其困难的。更重要的是，正如参考文献［397］中所指出的，若由于可利用的数据很少而选择简单的模型，这种做法意义也不是很大。然而，简单模型仍被广泛应用，而且有时效果也很好。但实际上，数据量的多少与数据源的复杂度完全无关。不难想像一个非常复杂的数据源可能只拥有相对较少的数据量的情况。因此即使在数据十分缺乏的情况下，也不能用机器学习方法来代替先验知识。但在任何情况下，推理和归纳始终是机器学习和计算生物学的中心问题。

在进行确定性推理时，人们使用演绎的方法。所以在诸如物理、数学等信息贫乏的学科中，最高级的一些理论都表述为公理体系。演绎法不会产生争议。绝大多数人都认可使用以下的特定方式进行演绎：如果X能推出Y且X为真，则Y必须为真。这是布尔代数的本质，也是所有数字计算机的基础。而在存在不确定性的情况下进行推理，常使用归纳和推断的推理方法：如果X能推出Y且Y为真，则X极有可能为真。有一组简单的特定规则用于归纳、模型选择和比较，这一方法称为贝叶斯统计推断，对于这个令人惊讶的方法，人们至今了解较少。贝叶斯方法已经存在一段时间了，但是直到最近才开始系统地渗透到科学和技术的不同领域，并取得了有用的成果。[229,372,373] 尽管在有些人看来，机器学习只是模型和学习算法的“电子化联合体”，但我们相信贝叶斯体系为不同算法技术的统一提供了一个坚实的理论基础。下面将对贝叶斯体系做一个简要概述。在后面的章节中，我们将这一体系应用于一些特定类型的模型和问题。

可以简单直观地描述贝叶斯方法的思想。贝叶斯方法对任意命题、假设或模型都赋予了一个似真度。（本书中“假设”和“模型”在本质上是同义的，但“模型”倾向于包含带许多参数的复杂“假设”。）具体地说，要合理地实现归纳过程，应该遵循以下三个步骤：

1. 清楚地描述出假设或模型，包括所有背景信息和数据。
2. 使用概率论的语言赋予假设一个先验概率。
3. 在推断过程中使用概率计算，特别是根据已知数据估计假设的后验概率（或者置信度），得到惟一的解。

这一种方法看起来当然是合理的。注意贝叶斯方法并不直接关注新的假设或模型产生的过程，它只关注利用已有的知识和数据对模型进行评价。而这种评价

过程可能对产生新的思想很有帮助。

但是为什么贝叶斯方法如此引人注目？为什么使用概率论的语言描述，而不用其他的方法？这是因为从严格的数学意义上说，它是进行不确定性推理的惟一一致的方法。这个回答令人吃惊。特别是有一组非常简单的常识性公理，即考克斯—杰恩斯公理（Cox Jaynes axioms），可以证实贝叶斯方法是推断和归纳的惟一一致的方法。根据考克斯—杰恩斯公理，似真度完全满足所有的概率规则。因此，推断、模型选择和模型比较所需要的仅仅是概率运算。

在下一节中，我们将用考克斯—杰恩斯公理给出贝叶斯思想的一个简单的公理化描述。为了简便起见，我们没有给出贝叶斯方法的任何证明和历史背景，也没有讨论任何与统计学基础有关的存在争议的问题。所有这些都可以在不同的书籍和文章中找到，如参考文献［51,63,122,433,284］。

2.2 考克斯—杰恩斯公理

我们在统计推断中处理的对象是关于客观世界的命题。例如，一个典型的命题X是“字符A出现在序列O的第i个位置。”一个命题不为真即为假，我们用$\bar{X}$表示命题X的否命题。一个关于客观世界的假设H也是一个命题，虽然它可能是一个由许多基本命题组成的复杂命题。模型M也可看做一个假设，不同之处在于模型通常包含带大量参数的复杂假设。在参数十分重要的情况下，我们将假定$M=M(w)$，其中w是所有参数组成的向量。一个复杂的模型M可以很容易地简化为一个二值命题，形式为“使用模型M解释数据D，误差率为ε”（在后面的讨论中，这个模糊的陈述将变得更加精确）。但在下面的论述中，不再对术语“模型”和“假设”加以区分。

虽然命题非真即假，我们仍然需要在不确定性存在的情况下进行推理。因此，给定一定量的信息I后，我们可以将每一个假设和一个似真度或置信度（也称为可信度或可靠性）联系起来，用符号$\pi(X|I)$表示。$\pi(X|I)$在这里只是一个符号，很显然，想要得到一个科学的论述，我们必须对置信度进行比较。即对任意两个命题X和Y，我们或者认为X比Y可信，或者认为Y比X可信，或者认为二者同样可信。我们用符号“>”表示这种关系，因此，如果X比Y可信则记为$\pi(X|I)>\pi(Y|I)$。不可否认，要想使这种比较更加切合实际，关系“>”应该具有传递性。也就是说如果X比Y可信，而Y比Z可信，则X一定比Z可信。于是我们得到第一个公理：

$$\pi(X|I)>\pi(Y|I) \text{ 且 } \pi(Y|I)>\pi(Z|I) \text{ 蕴涵着 } \pi(X|I)>\pi(Z|I) \qquad (2.1)$$

这个公理是很显然的，它还有一个重要的推论：">"是一个序关系（ordering relationship），因此置信度可以用实数表示，即从现在开始$\pi(X|I)$表示一个数。这当然不意味着这个数很容易计算，只是说明这个数是存在的，并且假设之间的序关系可以用这些实数的序关系表示。为了更深入地讨论和计算置信度，我们还需要一些另外的公理或规则，这些公理和规则便于合理地使用实数来表示置信度。

令人吃惊的是，我们只需要另外两个公理便足以把我们的理论完全确定下来。这个公理化的描述归功于考克斯和杰恩斯。[138,283] 为了更好地理解这两个公理，可以设想一个由简单开关组成的世界，在每一时刻某个给定的开关不是开就是关。因此，在某一特定时刻，这个世界中所有的基本假设或命题都有一个简单的形式"开关X开"或"开关X闭"。（对于序列分析问题，读者可以认为开关X的开、闭与字符X是否存在相对应，这并不妨碍对内容的理解。）显然，如果我们觉得"开关X为开"（命题X）更可信，那么"开关X为闭"（命题$\overline{X}$）就不那么可信。因此，对于任一给定的命题X，$\pi(X|I)$和$\pi(\overline{X}|I)$之间应存在某种关系。无需对这种关系进行任何假设，可以合理地认为这种关系对于所有开关和所有类型的环境信息，即所有命题X和I都是相同的。因此，可以用数学形式将第二个公理描述为：*存在一个函数F使得*

$$\pi(\overline{X}|I)=F[\pi(X|I)] \tag{2.2}$$

第三个公理要稍微复杂一些。考虑两个开关X和Y，它们相应地有四种可能的组合状态。那么，以X开Y闭的情况为例，其置信度依赖于开关X开的置信度，以及在已知开关X开的情况下开关Y为闭的置信度。同样地，这种关系不依赖于具体开、关的是哪个开关，也不依赖于各种环境信息。因此，可以用数学形式将第三个公理描述为：存在一个函数G使得

$$\pi(X, Y|I)=G[\pi(X|I), \pi(Y|X, I)] \tag{2.3}$$

迄今为止，我们并没有过多地讨论信息I。I对应于所有可用的信息共同构成的一个联合命题。I可以表示背景知识，例如生物大分子的一般结构或功能信息。I也可以是特殊的实验数据或其他数据信息。如果需要集中考虑一个特定的数据集D，可以记为$I=(I, D)$。在任何情况下，I可以是不固定的，它可以由任意多个代表命题的符号进行扩充或替换，正如（2.3）的右端所表示那样。例如，如果数据是以顺序的方式获得的，可记为$I=(I, D_1, \cdots, D_n)$；如果I是被明确定义并且是固定的，则完全可以将它从这些公式中去掉。

计算置信度的方法由上面提到的三个公理完全确定。特别地，我们可以证明存在一个置信度比例因子κ，使得$\mathbf{P}(X|I)=\kappa(\pi(X|I))$在[0,1]之间。更进一步，存在一个$\kappa$使得$\mathbf{P}$是惟一的，并且满足所有概率理论规则。若置信度严格地在区间[0,1]上取值，那么函数F和G一定由$F(x)=1-x$和$G(x,y)=xy$给出。这里不再给出相关的证明，有兴趣的读者可以参阅文献[138,284]。从而，第二个公理可以改写为概率的加法规则：

$$\mathbf{P}(X|I)+\mathbf{P}(\bar{X}|I)=1 \tag{2.4}$$

第三个公理可以改写为乘法规则：

$$\mathbf{P}(X,Y|I)=\mathbf{P}(X|I)\,\mathbf{P}(Y|X,I) \tag{2.5}$$

由此，我们可以用概率来代替置信度。值得注意的是，如果不存在不确定性，即$\mathbf{P}(X|I)$等于0或1时，那么作为特殊情况，当对命题进行求“反”和求“与”时，从（2.4）和（2.5）可以推导出演绎或布尔代数的两个基本规则：（1）“X或$\bar{X}$”总为真；（2）当且仅当X和Y都为真时，“X与Y”为真。使用对称性$\mathbf{P}(X,Y|I)=\mathbf{P}(Y,X|I)$和（2.5），可得到重要的贝叶斯定理：

$$\mathbf{P}(X|Y,I)=\frac{\mathbf{P}(Y|X,I)P(X|I)}{\mathbf{P}(Y|I)}=\mathbf{P}(X|I)\frac{\mathbf{P}(Y|X,I)}{\mathbf{P}(Y|I)} \tag{2.6}$$

贝叶斯定理十分基本，因为它允许互换条件项和非条件项。从某种意义上说，由于它确切地描述了如何根据Y所提供的新的信息修正X的置信度$\mathbf{P}(X|I)$，从而得到新的$\mathbf{P}(X|Y,I)$，因而它是推理或学习的过程。$\mathbf{P}(X|I)$也称为先验概率，而$\mathbf{P}(X|Y,I)$称为给定Y的后验概率。若能够不断地补充新的信息，这个规则显然是可以迭代的。在本书中，$\mathbf{P}(X)$一般表示X的概率。然而要明确一点，X的概率依赖于上下文，它显然不是一个通用的概念，它受背景信息的性质以及所考虑的候选假设空间的影响。

最后，我们应该意识到存在一组更普遍的公理，支持一个包括贝叶斯概率理论在内的更完备的理论体系。这就是关于决策或效用理论的公理体系，该体系注重怎样在存在不确定性的情况下得到“最优”的决策（参见附录A）。[238,63,431]无疑地，构成决策理论的简单公理可以帮助人们构造和评价与不确定的环境相联系的贝叶斯概率，并使相关的期望收益最大。事实上，一个更具一般性的理论是博弈论，该理论的不确定环境包括了其他的代理人或博弈者。由于本书的重点是数据建模，所以并不需要这些更一般性的公理化理论。

2.3 贝叶斯推断和归纳

下面讨论我们最感兴趣的统计推断：由一组数据D导出一个参数化模型$M=M(w)$。为了简化问题，下面的等式不再给出背景信息I。由贝叶斯理论我们立即可以得到

$$\mathbf{P}(M|D)=\frac{\mathbf{P}(D|M)\mathbf{P}(M)}{\mathbf{P}(D)}=\mathbf{P}(M)\frac{\mathbf{P}(D|M)}{\mathbf{P}(D)} \tag{2.7}$$

先验概率$\mathbf{P}(M)$表示在没有得到任何数据之前所估计的模型M为真的概率。后验概率$\mathbf{P}(M|D)$表示我们观测到数据集D后重新计算的模型M为真的概率。$\mathbf{P}(D|M)$是指似然度。

对于顺序获得的数据，我们有

$$\mathbf{P}\left(M\middle|D^1;\cdots,D^t\right)=\mathbf{P}\left(M\middle|D^1;\cdots,D^{t-1}\right)\frac{\mathbf{P}\left(D^t\middle|M,D^1;\cdots,D^{t-1}\right)}{\mathbf{P}\left(D^t\middle|D^1;\cdots,D^{t-1}\right)} \tag{2.8}$$

换言之，修正前的后验概率$\mathbf{P}(M|D^1,\cdots,D^{t-1})$成为新的先验概率。出于技术的原因，这些概率有可能非常小，处理相应的对数要容易得多，因此

$$\log\mathbf{P}(M|D)=\log\mathbf{P}(D|M)+\log\mathbf{P}(M)-\log\mathbf{P}(D) \tag{2.9}$$

为了将公式（2.9）用于任何种类的模型，我们需要具体说明先验概率$\mathbf{P}(M)$和数据似然度$\mathbf{P}(D|M)$。一旦先验概率和数据似然度定义清楚了，初始的建模工作就完成了，剩下的就是运用概率进行计算。但在此之前，让我们先简单地考察一些与先验概率和似然度有关的常见问题。

2.3.1 先验概率

使用先验概率是贝叶斯方法的一个优势，因为它允许将先验知识和约束条件导入建模过程。由于先验概率带有主观性，而且不同的先验概率会得到不同的结果，所以有时这个优势也被视为贝叶斯方法的缺点所在。针对这些反对意见，贝叶斯学派至少能够提供四种不同的解答：

1. 通常说来，随着数据量的增加，先验概率的作用减少。在形式上，这是因为负对数似然度$-\log\mathbf{P}(D|M)$随着D中数据量的增加呈线性增长，而先验概率$-\log\mathbf{P}(M)$保持不变。
2. 在有些情况下，可以使用一些客观准则，如最大熵（MaxEnt）、群不变性

（group invariance）等来确定无信息的先验概率值（见参考文献［228］）。

3. 甚至当没有明显提到先验概率时，他们也被隐含地使用了。贝叶斯方法在解决问题时，不一定需要揭示出隐含的先验概率问题。
4. 最后，也是最重要的，与不同模型及模型类别一样，不同的先验概率的影响可以在贝叶斯体系内通过比较相应的概率进行评估。

在统计学界有一个争论，即是否存在一个对所有情况都适用的决定先验概率的普遍客观原则，最大熵是否就是这样的一个原则。正如在附录B的最后部分简单讨论的那样，我们认为这种普遍原则其实并不存在。由于选择某种先验分布以及相应的数值结果一直隐含在整个概率计算过程中，我们最好抱着尝试的态度去选择先验分布。然而最大熵在一些特定的场合还是有用的。为了完整起见，现在我们简单概述一下如何依靠最大熵原则和群理论来确定先验概率，并介绍三种广泛使用的先验分布。

最大熵

最大熵原则规定先验概率应该是与所有先验知识或约束一致的概率分布中熵值最大的那个概率（信息理论的所有概念，如熵和相对熵的完整介绍，可参考附录B）。因此得到的先验分布是“最少假设的”，“最大模糊的”，或具有“最大不确定性”。根据拉普拉斯“无差别原则”，缺乏先验约束将导致均匀分布。因此，除了参数w的范围以外没有关于w的可用信息时，在此范围上的先验概率呈均匀分布是非常自然的选择。应用在建模中的最大熵可以由分布P或相应的直方图确定。最大熵等价于使用熵先验概率$\mathbf{P}(P)=e^{-\mathcal{H}(P)}/Z$，其中$\mathcal{H}(P)$是$P$的熵。我们将在3.2节中应用最大熵并对其做进一步讨论。最大熵也可看做更一般化的熵概念，即最小相对熵的一个特例（参见附录B）。[486]

群论的讨论

在许多情况下，先验分布的某些约束条件可以用群论的术语来表达，例如变换群的不变性。一个典型的例子就是标度参数（scale parameter），例如高斯分布的标准方差σ。假定我们只知道σ的范围，表示为$e^a<\sigma<e^b$。那么在此范围内，当σ变化时，它的密度$f(\sigma)$不变，因此f应与$d\sigma/\sigma$成正比。经过简单的归一化，可以得到

$$f(\sigma)=\frac{1}{b-a}\frac{d\sigma}{\sigma} \tag{2.10}$$

这与$\log\sigma$在区间［a, b］上均匀分布，或σ和σ^m的密度相同是等价的。群不变性分

析的其他例子可见参考文献［282,228］。

常用的先验分布：高斯分布、伽玛分布和Dirichlet分布

当先验概率分布不是均匀分布时，对于连续变量有两个实用的标准先验分布：高斯（或正态）先验分布和伽玛先验分布。均值为0的高斯先验分布常被用来初始化神经网络中各单元之间的权重。单参数的高斯先验分布形式如下：

$$\mathcal{N}(w|\mu,\sigma)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(w-\mu)^2}{2\sigma^2}\right) \tag{2.11}$$

联系上文，高斯分布具有优越性的原因之一是它与最大熵原则相关联。当可用的信息只有连续分布密度的均值μ和方差σ^2时，高斯密度$\mathcal{N}(\mu,\sigma)$可以达到最大熵（参见附录B）。[137]

具有参数α和λ的伽玛密度，在$w>0$时形式如下：

$$\Gamma(w|\alpha,\lambda)=\frac{\lambda^{\alpha}}{\Gamma(\alpha)}w^{\alpha-1}e^{-\lambda w} \tag{2.12}$$

其他情况下，该密度均为0。其中$\Gamma(\alpha)$是伽玛函数$\Gamma(\alpha)=\int_0^{\infty}e^{-x}x^{\alpha-1}dx$。通过调整$\alpha$和$\lambda$，并改变$w$，伽玛密度可以有一个很大范围的先验分布，并且能将密度集中在参数空间的一个特定区域内。当参数变量单边有界时，例如在标准差（$\sigma\geqslant 0$）为正参数的情况下，伽玛先验分布是十分有用的。

最后，对于在本书中起着非常基本作用的多项分布（例如在序列的指定位置上从字符集中选出一个字符），一个重要的先验分布类别是Dirichlet先验分布。[63,376]具有参数α和向量$Q=(q_1,\cdots,q_K)$的概率向量$P=(p_1,\cdots,p_K)$的Dirichlet分布具有如下形式：

$$D_{\alpha Q}(P)=\frac{\Gamma(\alpha)}{\prod_i\Gamma(\alpha q_i)}\prod_{i=1}^{K}p_i^{\alpha q_i-1}=\prod_{i=1}^{K}\frac{p_i^{\alpha q_i-1}}{Z(i)} \tag{2.13}$$

其中α，p_i，$q_i\geqslant 0$并且$\sum p_i=\sum q_i=1$。对于这种Dirichlet分布，存在$\mathbf{E}(p_i)=q_i$，$\mathbf{Var}(p_i)=q_i(1-q_i)/(\alpha+1)$，以及$\mathbf{Cov}(p_i p_j)=-q_i q_j/(\alpha+1)$。因此该分布的均值为$Q$，$\alpha$决定了该分布曲线在均值附近的光滑程度。Dirichlet先验分布之所以重要，是因为它们是多项分布的自然共轭先验分布，这一点将在第3章中加以证明。这意味着在具有Dirichlet先验分布的多项分布中，观察一些数据后得到的后验参

数分布也具有Dirichlet分布的形式。Dirichlet分布可以看做贝塔分布向多维的推广，它也可被解释为分布P的空间上的最大熵分布，它满足关于平均距离的约束条件，这个平均距离是该分布与由Q和α决定的参考分布之间的距离（如相对熵定义的距离，参见附录B）。

2.3.2 数据似然度

要想定义$\mathbf{P}(D|M)$，必然要掌握如何从模型M得到一个不同的观测集D'：在贝叶斯体系中，序列模型必须是概率形式的。除了自己产生的数据外，确定性的模型赋予其他数据的概率都为0。这在生物学上显然是不合适的，这可能也是由贝叶斯思想产生的一个主要的教训。如果没有如实给定似然度，序列模型的科学论述——它们如何拟合数据以及它们之间怎样进行比较——是不可能的。

似然度问题显然与变异及噪声问题相关。生物序列本身就存在噪声，而进化使随机事件的效应放大，最终导致变异。特定个体序列与一个家族（如一个蛋白质家族）中的“平均”序列之间不匹配和存在差异，这是必然的，并且必须进行量化。因为同样的DNA序列或氨基酸序列在同一物种的不同个体之间是不同的，在不同物种之间这种差异会更大，所以经常需要用概率的观点来考虑模型。事实上，过去使用的大量模型或多或少使用了启发式的方法，却没有清晰地指出其概率意义。一旦他们的概率意义表述清楚，这些方法的实际作用就一目了然了。从概率角度进行处理不仅仅澄清各种问题，使论述更严格，还能经常提示新的建模方法。

似然度的计算当然是与模型相关的，无法就其一般性进行讨论。在2.4节中，我们将给出一些用于推导模型的一般性原则，其中似然度的计算并不困难。但是读者需要认识到，不论使用什么准则来衡量模型和数据之间的差异或误差，这些准则都必须是来自一个基本的概率模型，而且需要明确地给出这一基本的概率模型，并使之符合贝叶斯分析的检验。事实上，如果具有参数w的模型$M=M(w)$由某个需要最小化的误差函数$f(w, D)\geqslant 0$来评价，那么相关的似然度可以定义为

$$\mathbf{P}\left(D|M(w)\right)=\frac{e^{-f(w,D)}}{Z} \qquad (2.14)$$

其中$Z=\int_w e^{-f(w,D)}dw$是保证概率积分为1的归一化因子［统计力学里的分割函数（partition function)］。结果，最小化误差函数与最大似然估计（ML）等价，或者更广义地，与最大后验概率（MAP）估计等价。特别是，当计算误差的平方和

来进行参量比较时（这是一种常见的做法），意味着将模型默认为一个高斯模型。这里以贝叶斯观点阐述了概率假设的新含义，这必将成为指导模型与数据匹配的准则的基础。

2.3.3 参数估计和模型选择

现在我们来看通常的贝叶斯推断方法。可以通过比较两个特定的模型M_1和M_2的概率$\mathbf{P}(M_1|D)$和$\mathbf{P}(M_2|D)$来比较这两个模型。通常的目的之一就是要发现或估计出某一组模型中的“最优”模型，即：寻求一组参数w，使得后验概率$\mathbf{P}(M|D)$或$\log \mathbf{P}(M|D)$最大，而相应的误差最小（参见附录A）。这就是MAP估计。由于正值便于处理，也使用等价的最小化$-\log \mathbf{P}(M|D)$：

$$\mathcal{E}=-\log \mathbf{P}(M|D)=-\log \mathbf{P}(D|M)-\log \mathbf{P}(M)+\log \mathbf{P}(D) \tag{2.15}$$

从优化的观点来看，先验概率的对数起到了正则因子（regularizer）的作用，正则因子就是一个附加的惩罚项，它可以用来体现附加的约束，如平滑性等。注意，(2.15)中的$\mathbf{P}(D)$起到归一化常数的作用，它不依赖于参数w，因此也与优化无关。如果对于所有考虑的模型，先验概率$\mathbf{P}(M)$取相同值，那么问题就简化为寻找$\mathbf{P}(D|M)$或$\log \mathbf{P}(D|M)$的最大值，这就是ML估计。总而言之，本书以及机器学习应用中的大部分内容都是基于MAP估计的，即最小化下式：

$$\mathcal{E}=-\log \mathbf{P}(D|M)-\log \mathbf{P}(M) \tag{2.16}$$

或更简单的ML估计，即最小化下式：

$$\mathcal{E}=-\log \mathbf{P}(D|M) \tag{2.17}$$

在最有意思的模型中，需要优化的函数十分复杂，其形式不能通过解析求解。因此需要采取迭代或随机方法，如梯度下降法（gradient descent）或模拟退火算法（simulated annealing），而最后得到近似的或次优的解。

贝叶斯推断采用迭代形式。第一步是在模型类中找到满足约束条件的最有可能的模型，然后再从中寻找最优解。但需要注意的是，仅当概率分布$\mathbf{P}(M|D)$的最优解惟一且周围的曲线形状很尖锐时，该最优解才真正有效。在不确定性很高并且可用数据相对较少的情况下，这一方法就不适用了。因此，贝叶斯方法对于模型全空间上的函数$\mathbf{P}(M|D)$（而并不只是最大值）更感兴趣，准确地说，贝叶斯方法是估计$\mathbf{P}(M|D)$的期望值。因此对于预测问题、多余参量的边缘化以及类别比较之类的问题需要进行更高级的贝叶斯推断。

2.3.4　预测、多余参数的边缘化和类别比较

考虑如下的预测问题：对于一个未知的参数化函数f_w，给定一个输入x，想要预测其输出y。很容易证明，其最优预测由下面的期望给出

$$\mathbf{E}(y)=\int_w f_w(x)\mathbf{P}(w|D)dw \tag{2.18}$$

这个积分是每个可能的模型 f_w 得到的预测值被各自的置信度$\mathbf{P}(w|D)$加权后的平均。另一个例子是边缘化过程，其中后验概率参数分布的积分仅对参数的一个子集进行，这个子集中的参数被称为多余参数。在概率论体系中，概率被定义为被观测到的频率，而参数分布的概念并没有定义，因此无法计算多余参数的积分。最后，人们也常遇到两个模型类C_1和C_2的比较问题。为了在贝叶斯体系中比较C_1和C_2，我们必须利用贝叶斯理论计算$\mathbf{P}(C_1|D)$和$\mathbf{P}(C_2|D)$：$\mathbf{P}(C|D)=\mathbf{P}(D|C)\mathbf{P}(C)/\mathbf{P}(D)$。除了先验概率$\mathbf{P}(C)$，还必须通过所有模型类的平均来计算"实测概率"$\mathbf{P}(D|C)$：

$$\mathbf{P}(D|C)=\int_{w\in C}\mathbf{P}(D,w|C)dw=\int_{w\in C}\mathbf{P}(D|w,C)\mathbf{P}(w|C)dw \tag{2.19}$$

相似的积分形式还出现在层次模型（hierarchical model）和超参数中（详见下文）。当似然度$\mathbf{P}(D|w,C)$在其最大值附近形成形状较尖锐的峰时，利用这个模型，期望值可以用最大概率值近似。但是通常来说，使用（2.18）和（2.19）的积分形式，需要更好的近似法——例如使用蒙特卡罗采样法，[491,396,69]这将在第4章中简要介绍。然而这些方法的计算量很大，而且并不适用于所有考察的模型。本书最关注似然度计算和第一层次的贝叶斯推断（ML和MAP）。更高层次的推断方法的研究十分活跃，这些方法应该在任何可能的情况下被考虑到。在更高层次的推断方法中，可能的计算能力当然是一个重要的因素。

2.3.5　奥卡姆剃刀原则

正如2.1节最后指出的那样，在可用的数据不充足的基础上选择简单的模型是没有意义的。但是在其他因素都相同的情况下，人们应尽量选用简单的假设，而非复杂的。这就是奥卡姆剃刀原则（Ockham's razor）。正如一些研究者指出的，奥卡姆剃刀原则至少通过两种方式自动体现在贝叶斯体系中。[285,373]首先，人们很容易选择那些对复杂模型进行惩罚的先验概率。即便没有这些先验概率，参数化的复杂模型也倾向于与数据量大的空间相一致。由于似然度$\mathbf{P}(D|M)$在数据空间上的和必须为1，如果$\mathbf{P}(D|M)$覆盖了数据空间的大部分，那么这个数据集的

似然度的平均值将较小。因此对于观察数据，在其他条件一致的情况下，复杂模型相应地得到较小的似然度。

2.3.6 最小描述长度

另一种建模方法是最小描述长度（minimum description length, MDL）。[446] MDL与数据压缩及信息传输的概念有关，它的目标是通过通信信道传输数据。"如实"传输数据并不经济：非随机的数据包括结构和冗余，因此必须进行压缩。好的数据模型应该抓住数据的结构特征并进行有效的压缩。最优的模型压缩是使描述数据所需的总信息长度最小的模型。它包括了指定压缩模型所需的长度和经模型压缩后数据的长度。通常，MDL与贝叶斯思想密切相关。根据香农的通信理论，具有概率p的事件的通信所需信息的长度与$-\log p$成正比。[483] 因此最可能的模型具有最小的描述长度。MDL和贝叶斯思想之间存在一些细微的差别，但这些不是我们关心的问题。

2.4 模型结构：图模型及其他技巧

显然，构造或选择合适的模型是由数据集决定的，同时还与建模者的经验和创造性有关。这里，我们只能重点讨论决定模型结构的一般性技术和技巧中的一小部分。文献中的大部分模型可以看做这些简单技术的组合应用。由于在机器学习中，贝叶斯分析的出发点通常是高维的概率分布$\mathbf{P}(M, D)$，相关的条件分布和边界分布［后验概率$\mathbf{P}(M|D)$、似然度$\mathbf{P}(D|M)$、先验概率$\mathbf{P}(M)$和事实概率$\mathbf{P}(D)$］，这些技术和技巧可以被视为分解、简化和参数化这些高维分布的不同方式。

2.4.1 图模型和独立性

迄今为止，最常使用的简化技巧是假定变量之间存在某种独立性，或者更准确地说，这种独立性是变量的子集关于其他给定子集的条件独立性。这些独立性关系常常可以用图来表示，图中的变量用节点来表示；而如果两个节点之间没有连接，则表示它们之间在某种程度上是独立的（准确的定义可以参见附录C）。关于这一问题的综述、处理方法和相关文献的索引见参考文献［416,350,557,121,499,106,348,286］。

独立性关系导致了一个基本事实：所有变量的全局高维概率分布可以化为几个低维空间的简单局部概率分布的乘积。这些低维空间是由较低水平的变量聚类

得到的，这些聚类显示在图的结构中。

根据图中的边是否有向，图模型可以分为两个主要的类别。无向边在相互关系对称的问题中十分典型，例如统计力学或图像处理问题。[272,199,392]在无向图的情况中，这些模型常常被称为马尔可夫随机场（Markov random field），无向概率独立性网络，波耳兹曼机（Boltzmann machine），马尔可夫网络（Markov network）和对数线性模型。

有向模型常用于相互关系不对称的问题，可以反映因果关系或时间的不可逆性。[416,286,246]这在专家系统和所有基于时间数据的问题中十分典型。在信号处理和控制中广泛使用的卡尔曼滤波器（Kalman filter）可以看做属于这种体系。在时间序列中，独立性假设常常用在马尔可夫模型中。本书中讨论的大部分模型——特别是神经网络模型（NN）和隐马氏模型（HMM）——都是有向边的图模型的例子。生物信息学中图模型的系统性应用将在第9章中介绍。这类典型模型包括：贝叶斯网络，置信网络，有向概率独立性网络，因果网络以及影响图（influence diagram）。对于混合的情况也有可能发展一套理论，模型中有向边和无向边都存在。[557]这种混合的图形也称做链式独立性图。图模型的基本理论参见附录C。

这里我们介绍在后面的章节中要用到的一些符号表示方法。图G由$G=(V, E)$表示，其中集合V表示顶点，集合E表示边。如果这些边是有向的，我们就记为$G=(V, \vec{E})$。在无向图中，$N(i)$表示顶点i的所有相邻顶点的集合，$C(i)$表示所有与i连通的顶点的集合。因此有

$$N(i)=\{j\in V:\ (i,j)\in E\} \tag{2.20}$$

在有向图中，我们用明显的记号$N^-(i)$和$N^+(i)$分别表示i的所有父节点和所有子节点。类似地$C^-(i)$和$C^+(i)$分别表示i的祖先（或“过去”）和后代（或“将来”）。所有这些记号都可以扩展到任意顶点的集合I。所以对于任意$I\subseteq V$，有

$$N(I)=\{j\in V:\ \exists i\in I\quad (i,j)\in E\}-I \tag{2.21}$$

这也称做是I的边界。

一个基本的现象是，在许多应用中所产生的图经常是稀疏的（sparse）。因此全局概率分布可以分解为少量的几个较小的局部概率分布。这是实现用于学习和推断的有效计算结构的关键，这种计算结构基于信息在图的各个变量类中的局部传播。接下来的一些技术基于图模型的一般思想，但也常常会被单独提出来加以研究。

2.4.2 隐变量

在许多模型中，一个典型假设是：数据部分地由隐含（hidden）或潜在（latent）变量生成，而这些变量或者不包含在数据记录中，或者根本不可观测。[172] 缺失的数据也可以看做隐含的变量。网络中的隐节点输出或HMM的状态链都是典型的隐变量（hidden variable）。另一个隐变量的例子则是混合模型的系数（详见下文）。显然，模型的参数，例如神经网络的权重或HMM的生成/转移概率，在某种意义上也可以视为隐变量，尽管这与传统的术语形式不符。隐变量模型中的典型推断问题是估计隐变量集上的概率分布以及相应的期望值。这通常是大规模参数化模型（如HMM）的参数估计问题中的子问题。EM算法是对丢失数据的模型或隐变量模型进行参数估计的重要算法，我们将在第4章中详细介绍这一算法，并在第7章中进一步描述它在HMM中的应用。

2.4.3 层次模型

许多问题都有内在的层次结构或分解。这可能是由于问题中存在不同的时间标度或长度标度引起的。上面描述的图模型中的子集类，可以看做更高层次的表示数据结构的图模型的节点［例如参考文献［350］中关于交叉树(junction tree)的概念］。与之相关地，模型参数的先验概率也可以具有层次结构，其中某一层的参数可以递归地用于定义下一层参数的先验分布。随着模型层次的提高，模型的参数一般会有所减少。一个特定层上的所有参数常被称做是该层的“超参数”。

超参数能够在控制模型的复杂性和结构的同时提供更大的灵活性。超参数具有“高增益”(gain)，即超参数变量的微小变化可以导致其下的各层模型发生巨大变化。因为模型的先验概率可以通过一定数目（通常很少）的超参数计算，所以超参数也可以实现参数约简。先验概率的超参数计算法用符号表示，即

$$\mathbf{P}(w)=\int_{\alpha}\mathbf{P}(w|\alpha)\mathbf{P}(\alpha)d\alpha \tag{2.22}$$

其中α表示具有先验概率$\mathbf{P}(\alpha)$的参数w的超参数。一个典型的例子是神经网络中的连接权重。在一个给定的问题中，权重的先验分布一般使用均值为μ，标准差为σ的高斯分布。每一个权重都使用一组不同的超参数μ和σ，将使模型的约束大大减少。可以认为，在一个给定节点或一个完整层次上的所有σ是相同的。在更高的层次上，先验概率可由下层的几个σ确定，并依此类推。附录D中给出了一个分层的Dirichlet模型的例子。

2.4.4 混合建模/参数化

由于使用的模型规模通常都很大，参数化问题在机器学习中十分重要。即使数据和参数的全局概率分布已被分解为较小的局部概率分布的乘积，作为独立性假设的结果，仍常常需要对子模型的分布进行参数化。对分布进行参数化的两个常用的有效方法是混合模型和神经网络。

在混合模型中，一个复杂的分布P由一些简单或规范的分布的线性凸组合决定，其形式如下：

$$P=\sum_{i=1}^{n}\lambda_i P_i \tag{2.23}$$

其中$\lambda_i\geqslant 0$称为混合系数，它满足$\sum_i \lambda_i=1$。分布P_i称做混合分量，它们有自己的参数（均值、标准差等）。有关混合模型的概述见参考文献［173,522］。

神经网络也用来对模型进行重新参数化，即将模型参数作为输入和连接权重的函数进行计算。正如我们将要看到的，神经网络之所以有此功能的部分原因是它具有通用逼近特性和良好的灵活性，而且学习算法比较简单。最简单的例子可能就是回归问题，其中神经网络可以作为独立变量（输入）的函数，用来计算相关变量的均值。在第9章中会给出一个更精确的例子，其中神经网络用于计算HMM的生成参数和转移参数。有时可以用术语“混合”来描述不同模型类结合的情况，尽管这种结合可以采用不同的形式。

2.4.5 指数分布族

附录A简要介绍了指数分布族的相关概念。这里需要说明的是，许多最经常使用的分布（高斯分布，多项分布等）都属于这个系列，使用指数分布族中的某个分布常常可以得到十分有效的算法。指数族的概述和参考文献的全面列表见参考文献［94］。

2.5 小　结

上面简要地介绍了在建模和推断中使用的贝叶斯方法。贝叶斯方法的主要优点十分明显：它以坚实的概率论为基础，为统计推断提供了一套原则和灵活的方法。事实上，贝叶斯方法被广泛使用的一个重要原因是：它构建在一套规模非常小的公理集合之上并具有惟一形式。我们认为数学家可能比生物学家更容易接受

这个观点。

贝叶斯体系至少从三个不同层次阐明了一系列问题。首先，贝叶斯体系要求明确先验知识、数据和假设。贝叶斯体系质疑任何知识的不确定性，并且鼓励这种质疑。它处理建模过程中存在的内在主观性时，并不是简单地将其排除在模型之外，而是将其与建模过程相结合。从本质上来讲，这是一个推动模型不断精化的迭代过程。第二，也是最主要的，序列模型必须有概率意义，还可以用定量方法描述数据的变异和噪声。否则无法得到关于模型的严格科学描述，无法确定模型是否与数据相吻合，最终也无法对模型和假设进行比较。第三，贝叶斯体系阐明了如何进行推断，即如何利用概率比较不同的模型、量化误差和不确定性。特别地，它能够为提出的问题给出明确的、惟一的解答。它为客观建模定义了一组必要的规则。进行推断的最基本步骤是根据可用的数据和相应的期望，利用概率理论和数值估计规则计算模型的似真度。

贝叶斯方法可以引导我们更好地理解模型的弱点，进而帮助我们建立更好的模型。另外，随着生物大分子、结构、功能和调控模型数据的数量、范围和复杂性不断增加，很客观地比较不同的模型以及使用模型进行预测变得十分重要。随着数据库的规模和复杂性不断增加，模型比较和预测越来越成为中心问题。一些新思想很有可能在将贝叶斯体系引入序列分析的过程中萌发出来。

贝叶斯方法的主要缺点是它的计算量特别大，尤其是需要计算高维分布的均值时。仅就本书出现的最长序列而言，使用现有的任何计算机都无法得到其完整的贝叶斯积分。但这一问题会随着蒙特卡罗方法[491,69]和其他近似技术的不断发展，以及工作站和并行计算机计算能力的不断提高而逐步获得解决。

一旦建立起通用的概率体系，下一个研究重点将转移到关于图模型的讨论上，即利用独立性假设将高维概率分布分解并生成独立的子图。大多数机器学习模型和问题能够以其中所包含的变量（可观测的或隐藏的）和参数为基础，表示为递归的稀疏图形式。稀疏递归图（sparse recursive graph）可以很好地描述或表现绝大多数模型和机器学习方法。

第3章　概率建模和推断：应用举例

贝叶斯方法建模的本质是什么？显然对于任何一类模型，首先要明确地给出似然度$\mathbf{P}(D|M)$和先验概率$\mathbf{P}(M)$。本章中，我们将介绍广义概率体系的几个简单应用。第一个应用是基于掷骰子的简单序列模型。而本章中的其他应用，包括统计力学的基本推导，都是这个简单序列模型的变形推广，通过增加骰子数量或者改变观察的数据得到的。

3.1　最简单的序列模型

单个硬币投掷问题是最简单的，但并非不重要的一个建模实例。这个模型只有一个参数p以及一个由字符集A={H，T}中的字符组成的字符串数据集，其中H代表正面，T代表反面。由于我们的研究对象是DNA序列，因此直接将模型扩展到包含4个字符，并保证所观测到的字符串尽可能长。

3.1.1　序列数据的单骰子模型

数据集D由DNA链组成，每条链由字符集A={A，C，G，T}中的字符组合而成。我们想要使用的简单模型，假设字符串是通过独立投掷相同的四面体骰子得到的（图3-1）。

因为各次投掷相互独立，并且使用的骰子相同，从似然度的角度考虑，我们使用多个字符串还是一个较长的字符串没有区别。所以我们假设数据是一个长度为N：$D=\{O\}$的观察序列，其中$O=X^1\cdots X^N$，$X^i\in A$。模型M有四个参数：p_A，p_C，p_G，p_T，它们满足$p_A+p_C+p_G+p_T=1$。似然度由下式给出：

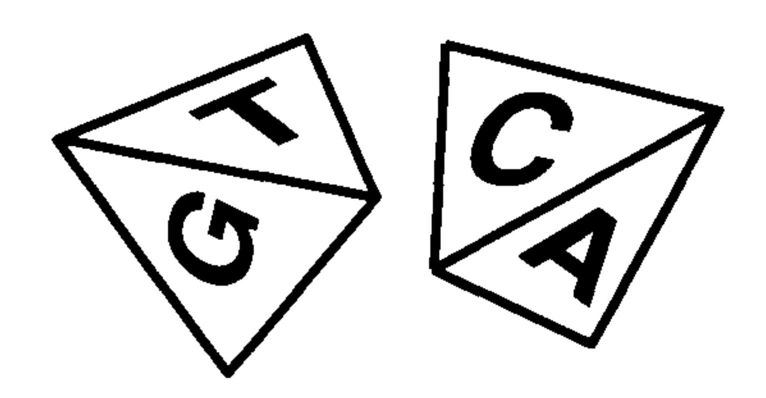


图3-1 用以产生DNA链的四面DNA骰子的两面

$$\mathbf{P}(D|M)=\prod_{X\in A}p_X^{n_X}=p_A^{n_A}p_C^{n_C}p_G^{n_G}p_T^{n_T} \quad (3.1)$$

其中，n_X是字符X在序列O中出现的次数。因此，负对数后验概率为

$$-\log\mathbf{P}(M|D)=-\sum_{X\in A}n_X\log p_X-\log\mathbf{P}(M)+\log\mathbf{P}(D) \quad (3.2)$$

如果我们假定所有参数具有均匀的先验分布，那么MAP参数估计问题就与ML参数估计问题等价，可以通过优化与负对数似然度相关的、带有归一化约束的拉格朗日算子进行求解：

$$\mathcal{L}=-\sum_{X\in A}n_X\log p_X-\lambda\left(1-\sum_{X\in A}p_X\right) \quad (3.3)$$

在这里以及本书的其他地方，我们直接在结果中检验是否满足正约束。令偏导数$\partial\mathcal{L}/\partial p_X$等于0，即可得$p_X=n_X/\lambda$。应用归一化约束得到$\lambda=N$，最后得到概率估计

$$p_X^*=\frac{n_X}{N}\text{，对于所有}X\in A \quad (3.4)$$

注意，对于最优的参数集P^*，当$N\to\infty$时，字符的平均负对数似然度应趋向于P^*的熵$\mathcal{H}(P^*)$（参见附录B）：

$$\lim_{N\to\infty}-\frac{1}{N}\sum_{X\in A}n_X\log\frac{n_X}{N}=-\sum_{X\in A}p_X^*\log p_X^*=\mathcal{H}(P^*) \quad (3.5)$$

从另一个角度看这一结果，除去熵值常数项之外，负对数似然度本质上是固

定的骰子概率p_X和观测频率n_X/N之间的相对熵。在下面有关统计力学的章节中，我们将看到它是如何与自由能的概念有关的。

当N很大时，用观测频率估计$p_X=n_X/N$是很自然的。大数定理告诉我们N值足够大时，观测频率将与真实的p_X值十分接近。但是当N很小（例如$N=4$）时又如何呢？假设在一个长度为4的序列中没有观察到字符A，我们是否要将概率p_A设为0呢？很可能不是，尤其在我们没有任何理由认为骰子是高度偏倚的情况下。换句话说，我们的先验知识并不认为于模型的参数值为0。正如在第2章中指出的，这种情况下相应的自然先验分布不应是均匀分布，而是关于参数向量P的Dirichlet先验分布。事实上，根据Dirichlet先验分布$\mathcal{D}_{\alpha Q}(P)$，负对数后验概率变为

$$-\log \mathbf{P}(M|D) = -\sum_{X \in A}[n_X + \alpha q_X - 1]\log p_X + \log Z + \log \mathbf{P}(D) \quad (3.6)$$

Z是Dirichlet分布的归一化常量，它不依赖于概率p_X。因此，除了n_X由$n_X+\alpha q_X-1$代替外，MAP最优化问题与前面解决的问题十分相似。（当其为正时）我们立即可以得到估计

$$p_X^* = \frac{n_X + \alpha q_X - 1}{N + \alpha - |A|}\text{，对所有}X \in A \quad (3.7)$$

特别地，Dirichlet先验分布的作用相当于在观测到的次数上增加一个"虚计数"（pseudocount）项。适当地选择平均化的分布Q（例如令Q为均匀分布）和参数α，可以使估计p_X^*的值总大于0。当Q为均匀分布时，Dirichlet先验分布是对称的。注意，P的均匀分布是对称的Dirichlet先验分布的一个特例，即$q_X=1/\alpha=1/|A|$。另外，由（3.6）也可清楚地看到，后验分布$\mathbf{P}(M|D)$是$\beta=N+\alpha$，$r_X=(n_X+aq_X)/(N+\alpha)$的Dirichlet分布$\mathcal{D}_{\beta R}$。

后验分布的期望是向量r_X，这与MAP估计［见（3.1）］有点不同。这意味着可以使用p_X的另一种估计，如预测分布或后验分布均值（mean posterior，MP）估计：

$$p_X^* = \frac{n_X + \alpha q_X}{N + \alpha} \quad (3.8)$$

这通常是一个比较好的选择。特别是MP估计可以最小化相对熵距离的期望值$f(P^*)=E(\mathcal{H}(P, P^*))$，其中期望值是针对后验概率$\mathbf{P}(P|D)$的。

单一Dirichlet先验分布的骰子模型十分简单，它使得人们可以用解析方法进

行更高层次的贝叶斯推断。例如，我们可以计算$\mathbf{P}(D)$：

$$\mathbf{P}(D)=\int\mathbf{P}(D|w)\mathbf{P}(w)dw=\int_{\sum p_{\mathrm{X}}=1}\prod_{\mathrm{X}\in A}p_{\mathrm{X}}^{n_{\mathrm{X}}+\alpha q_{\mathrm{X}}-1}\frac{\Gamma(\alpha)}{\Gamma(\alpha q_{\mathrm{X}})}dp_{\mathrm{X}} \tag{3.9}$$

这个积分式与Dirichlet分布的积分很相似，因此很容易通过计算得到

$$\mathbf{P}(D)=\frac{\Gamma(\alpha)}{\prod_{\mathrm{X}\in A}\Gamma(\alpha q_{\mathrm{X}})}\frac{\prod_{\mathrm{X}\in A}\Gamma(\beta r_{\mathrm{X}})}{\Gamma(\beta)} \tag{3.10}$$

这是先验概率分布和后验概率分布的归一化常数的比值。

读者可自行练习贝叶斯方法的应用。下面一些练习非常有用：寻找α和q_{X}的值使得（3.10）中的$\mathbf{P}(D)$最大，使用超参数定义α和q_{X}的先验概率，以及当先验分布是Dirichlet分布的混合时，研究MAP估计和MP估计，其中Dirchlet分布形式为：

$$\mathbf{P}(P)=\sum_i\lambda_i\mathcal{D}_{\alpha_i Q_i}(P) \tag{3.11}$$

（参见附录D和参考文献［489］）。在第二种MP估计的情况下，后验分布也是Dirichlet分布的混合。一个一般性的结论是：当先验分布是共轭分布的混合时，后验分布也是共轭分布的混合。

3.1.2 统计数据的单骰子模型

对于同样的骰子模型，我们假设可用的数据由字符出现的次数$D=\{n_{\mathrm{X}}\}$组成，而不是由实际的序列组成。一个简单的组合计算说明在这种情况下似然度具有以下形式

$$\mathbf{P}(D|M)=P(n_{\mathrm{X}}|p_{\mathrm{X}})=\frac{N!}{\prod_{\mathrm{X}\in A}n_{\mathrm{X}}!}\prod_{\mathrm{X}\in A}p_{\mathrm{X}}^{n_{\mathrm{X}}} \tag{3.12}$$

其中$\sum_{\mathrm{X}}n_{\mathrm{X}}=N$。这与（3.1）仅相差一个因子项，这个因子表示由字符集实现的长度为N的字符串的所有可能排列方式的数目（n_{X}）。这样一个由简单的骰子模型产生的计数为n_{X}的分布，也称做多项分布，是与投掷硬币（即两面的骰子）相关的二项分布的推广。虽然不甚恰当，骰子模型有时也称做多项模型。

如果参数向量P具有Dirichlet先验分布$\mathcal{D}_{\alpha Q}(P)$，相似的计算表明$P$的后验概率分布也是Dirichlet分布，记为$\mathcal{D}_{\beta R}(P)$，其中$\beta=N+\alpha$，$r_{\mathrm{X}}=(n_{\mathrm{X}}+\alpha q_{\mathrm{X}})/\beta$。相应

地，MAP估计和MP估计P^*也与（3.7）和（3.8）相同。

现在我们考虑特定向量P导出的次数n_X的分布。对（3.12）取对数并利用斯特林（Stirling）阶乘估计公式

$$n! \approx \left(\frac{n}{e}\right)^n \sqrt{2\pi n} \tag{3.13}$$

于是我们得到

$$\log(\mathbf{P}(D|P) \approx C - \mathcal{H}(n_X/N, p_X) \tag{3.14}$$

其中C是依赖于n_X的常量，$\mathcal{H}$是经验分布和P之间的相对熵。当P是均匀分布时，上述的相对熵可简化为经验分布的熵，其中只差一个常数项。因此，在这种情况下

$$\mathbf{P}(D|P) \approx \frac{e^{\mathcal{H}(n_X/N)}}{Z} \tag{3.15}$$

这称做熵分布。换言之，由均匀分布P可以导出一个关于次数n_X的、在所有可能直方图构成的空间之上的熵分布，正如我们将在3.2节中看到的，这是最大熵原则的主要初衷之一，相当于采用先验熵分布。注意Dirichlet分布和关于P的熵分布

$$\frac{\exp\left(-\sum_X p_X \log p_X\right)}{Z} \tag{3.16}$$

之间的相似性和不同。可以证明如果P具有熵分布，那么观察到n_X后得到的后验分布既不是熵分布，也不是Dirichlet分布，我们将这个证明作为练习留给读者。熵分布不是多项分布的共轭分布。如果具有先验熵分布，则MAP估计仍具有 $p_X^* = n_X/N$ 的形式。

尽管简单的骰子模型十分粗糙，但它正是我们计算1阶统计量时所使用的模型。这里所说的1阶统计量，即：一个给定的序列集合（如外显子、内含子或蛋白质家族）中每个字符出现的比例。这可以看做是一个迭代建模过程的第一步，因此后续模型的性能评价必然要考虑这个1阶模型。下一节的复合骰子模型和第7章中的隐马式模型（HMM）是简单骰子模型的推广。简单骰子模型可以通过将其每一面的字符变为字符串序列而得到一般推广。这与扩展字符集是等价的。例如，可以使用一个64面骰子构造DNA密码子的模型。

3.1.3　序列数据的复合骰子模型

另一个简单的序列模型是复合骰子模型。这里的数据由K个序列组成，每个

序列的长度都为N。例如，读者可以考虑一个K个序列的多重序列比对，其中间隙符号“-”可以看做字符集中的一个符号。在复合骰子模型中我们假定有N个独立的骰子，每个骰子对应一个位置，每个序列都是N个骰子按照一定的顺序投掷得到的结果。令 p_X^i 表示第i个骰子投出字符X的概率， n_X^i 表示在第i个位置上出现字符X的次数。由于假定骰子和序列都是独立的，似然度函数为

$$P(D|M)=\prod_{i=1}^{N}\prod_{X\in A}p_X^{n_X^i} \quad (3.17)$$

如果所有骰子均具有先验均匀分布，与单骰子情况相同，通过计算可以得到

$$p_X^{i*}=\frac{n_X^i}{K}\text{，对所有}X\in A \quad (3.18)$$

我们将Dirichlet先验分布对模型的影响以及可能的推广（见参考文献［376］）作为练习留给读者。n-gram模型是语言建模型中常用的一类著名模型。在n-gram模型中有$|A|^{n-1}$个骰子，每个骰子都与长度为$n-1$的不同前缀相关联，每个骰子都是有$|A|$个面的简单骰子，每个面对应1个字符。序列由一个长度为n的窗口扫描产生，选择与当前前缀相关的骰子随机投掷而得。因此对下一个骰子的选择依赖于先前投掷的结果。这种n-gram模型可以看做阶数等于前缀长度的马尔可夫模型，其中前缀也称做模型的“记亿”。单骰子模型的记亿长度为0。此外，还有可变记忆长度的变体模型（一个生物序列的应用例子见参考文献［448］）以及高阶混合马尔可夫模型，后者亦即插值马尔可夫模型。随着字符集的规模和记亿长度的增大，可能的前缀数目急剧增长，因此高阶马尔可夫模型对计算能力要求很高。然而由于DNA的字符集很小，所以仅使用阶数为5左右的马尔可夫模型仍是可行的。

3.2 统计力学

至少由于以下五个方面的原因，我们需要理解初步的统计力学及其与机器学习和计算生物学的联系。第一，尽管在我们看来，由于混淆了最大熵原理和贝叶斯推断，统计力学的常见表述有一些缺陷，但统计力学仍可被看做贝叶斯推断的最初也是最好的例子之一；[280,281] 第二，传统的统计力学考虑的是，如何从大量简单的微观相互作用单元中得到诸如平衡、相变等宏观统计性质；第三，统计力学的技术和结果对于理解机器学习中所使用的一系列图模型的性质和进化过程十

分有用；[252,482,50] 第四，统计力学模型也已直接应用到生物大分子上——例如蛋白质折叠问题（见参考文献［151］。第五，统计力学有利于理解机器学习的一些基本算法，例如第4章中描述的模拟退火算法和EM算法。

下面我们从最基本的贝叶斯统计估计开始推导统计力学的基本概念，特别是关于波耳兹曼-吉布斯分布和自由能的概念，这些将在下一章中用到。在基本的统计力学体系里，可以考虑一种包含大量微观状态$\mathbf{S}=\{s_1,\cdots,s_{|s|}\}$的随机系统，其中$p_s$表示对于给定分布$P=(p_s)$，系统处于状态$s$的概率。这可以被视为一个具有参数$w=p_s$的骰子模型$M(w)$，尽管不必假定骰子都是独立的。上述例子的主要区别就在于数据。骰子的各个面，即微观状态并不能由观测得到，只能作为隐变量处理。因此，我们假定存在一个关于状态的函数$f(s)$，惟一的宏观观测量（数据）就是f的均值或期望。在本节中我们将使用一些不很规范的符号用法，例如记$D=\mathbf{E}(f)=\sum_s p_s f(s)$。

统计力学中的状态经常有微观的结构$s=(x_1,\cdots,x_n)$，其中x_i是局部变量。例如，x_i可以是二值变量，在这种情况下$|S|=2^n$。同样，f是系统的能量，可以记为局部变量的二次函数：$f(s)=f(x_1,\cdots,x_n)=\sum_{ij}w_{ij}x_ix_j+\sum_i w_ix_i$。交叉参数（interaction parameter）w_{ij}可以是局部的（如点阵的旋转），也可以是全局的，并且与内在的图模型相关。尽管这一假设在特殊系统建模以及发展详细的理论时十分重要，但在后面的章节中并不需要用到它们。我们要问的第一个问题是：给定f的观测均值，关于状态分布P，我们可以得到什么结果？

3.2.1 波耳兹曼-吉布斯分布

标准推导

这方面的大部分标准处理都是基于最大熵原则的。不考虑任何附加信息，我们应选择满足约束$\sum_s f(s)p_s=D$并具有最大熵的分布P，因为这是最“全面”并且需要最少附加假设的解。这个问题可以很容易利用拉格朗日算子$\mathcal{L}$解决，其中$\mathcal{L}$是由需要优化的函数（带有相关约束条件）的线性组合：

$$\mathcal{L}=-\sum_s p_s\log p_s-\lambda\left(\sum_s p_s f(s)-D\right)-\mu\left(\sum_s p_s-1\right) \tag{3.19}$$

令$\mathcal{L}$对p_s偏导数为0，我们即可发现分布的惟一解具有如下形式：

$$p_s^*(\lambda)=\frac{e^{-\lambda f(s)}}{Z(\lambda)} \tag{3.20}$$

其中归一化因子$Z(\lambda)=\sum_s e^{-\lambda f(s)}$称做分割函数。在统计力学中，由定义$\lambda=1/kT$可知，拉格朗日算子与温度$T$有关，其中$k$是波耳兹曼常数。对于目前所有的问题，我们不需考虑温度，而是直接考虑参数λ。注意，从式子

$$\sum_s \frac{e^{-\lambda f(s)}}{Z(\lambda)} f(s) = D \tag{3.21}$$

可知λ以及相应的T，完全由观察值D决定，通常有必要假定$\lambda=1$。最优分布P^*称做系统的波耳兹曼-吉布斯分布。需要注意的是，至少对于给定的温度，使用与$-\log P$成比例的能量函数，任何分布P都可以表示为波耳兹曼-吉布斯分布。当参数p_s具有多重线性约束时，也很容易得到一个相似的公式。

尽管波耳兹曼-吉布斯分布非常有用，但从贝叶斯理论的观点来看，有三个原因使得标准推导过程不能令人完全满意：（1）先验分布不明确。因此，如何将有关p_s的附加先验信息（如已知第一个状态要比其他状态出现得更频繁）结合进来？（2）概率模型不明确，特别是如何计算似然度$\mathbf{P}(D|p_s)$。（3）最大熵的使用依据不足。尤其是最大熵与ML或MAP估计之间存在任何联系吗？实际上，最大熵的使用与前面所讨论的组合变量部分相关，即：当骰子相互独立时，熵的最大化本质上等价于可能实现的次数$N!/\prod_s n_s!$的最大化。[282]因此，最大熵的解就是那个能以最多种方法实现的解。这一论据只基于实现的次数，而没有考虑相关的概率。下面将针对这三个方面可能出现的问题进行讨论。

贝叶斯方法的推导

标准推导的主要问题是它的概率模型不明确。特别是，似然函数$\mathbf{P}(D|p_s)$没有明确的定义，而且如果不考虑系统的实际运行，在这一点上几乎得不到什么进展。因此我们必须增强初始设置，即假定存在一个给定的足够大的数字N，并假定已经对系统经过了一段时间的观察。我们还需考虑不同观察次数的影响，但这会使分析变得更复杂。因此，我们需要用观测次数n_s来参数化模型。注意实际观察到的是$D=(\sum_s n_s f(s))/N\neq\sum_s p_s f(s)$。

次数n_s可能有一些不同的先验分布。正如我们已经看到的那样，一个自然的先验分布是使用n_s/N的Dirichlet先验分布。非对称的Dirichlet先验分布可以很容易地结合与任何特殊状态出现频率有关的附加信息。我们将由Dirichlet先验概率得到后验概率的计算作为练习留给读者，显然这不是波耳兹曼-吉布斯解。例如，如果先验分布是均匀的，并且$f(s_1)=D$，那么向量（$N, 0,\cdots, 0$）具有最小的可能的熵，它可以确保数据的概率达到最大！这里我们使用熵分布作为先验分布，这是

P为均匀分布时得到的n_s的分布。另外，当所有过程独立时，这样的先验概率可以最好地确定，即这时的基本概率模型是简单的具有$|S|$面的骰子模型。尽管在接下来的讨论中我们只限于0阶的马尔可夫模型，读者可以自己考虑更高阶的马尔可夫模型。例如，1阶的马尔可夫模型将包括与状态之间的转移概率相关的一组不同的参数，这与$|S|$面骰子模型等价。第4章还将讨论1阶马尔可夫模型和波耳兹曼-吉布斯分布的一些问题。

由此似然函数变得平凡，它的值为1或0，这依赖于$D=\sum_s f(s)n_s/N$是否成立。我们可以将贝叶斯推断的第一步继续进行下去，由MAP估计估算出参数n_s。使用前面介绍的公式，即可得到拉格朗日算子

$$\mathcal{L}=-\sum_s \frac{n_s}{N}\log\frac{n_s}{N}-\lambda\left(\sum_s\left(f(s)\frac{n_s}{N}-D\right)\right)-\mu\left(\sum_s n_s-N\right) \tag{3.22}$$

其中熵起正则因子的作用。这实际上与式（3.19）是一样的，由此可以得到n_s/N的MAP波耳兹曼-吉布斯分布。用参数p_s代替n_s可以得到相似的结论，但是由于不同的n_s值均可与D相符，所以不论是确定熵先验概率还是计算似然函数都会变得复杂一些。

总之，波耳兹曼-吉布斯分布对应于应用MAP估计的贝叶斯推断的第一步，其中的先验分布为熵分布。因此，最好不要将最大熵看做普遍原则，而应该仅仅看做在多项分布情况下实现1阶贝叶斯推断的捷径，这个多项分布与熵先验相关。这一先验分布还有疑点：我们可以构造一些例子，其中最大熵原则会导出“错误”的解。我们将这种例子的构造以及更高阶贝叶斯推断的实现（计算超参数、综合先验概率）作为练习留给读者。

3.2.2　热力学极限和相变

温度是强度量（intensive quantity）的一个好例子，即该参数的定义与系统的规模无关。而与之相对的是广延量（extensive quantity），例如能量，它会随着系统规模的增大而增大。对于具有局部相互作用的大系统而言，这种增大与系统规模的增大呈线性相关。因此，当系统的规模趋于无穷大时，每一单位容量的广延量的值趋于一个有限值，称为热力学极限（thermodynamic limit）。

统计力学的一个主要目标就是估计宏观量的热力学极限，即根据波耳兹曼-吉布斯分布估计其期望值。特别地，一个主要的目标是得到分割函数$Z(\lambda)$的近似，因为这个函数包括了系统的大部分相关信息。尤其是我们很容易证明函数f的任何矩量都可以由$Z(\lambda)$计算得到，更准确地说，可以由$Z(\lambda)$的对数得到。

例如，对于头两个矩量，即均值和方差，可通过初等计算得出：

$$\mathbf{E}(f) = -\frac{\partial}{\partial \lambda} \log Z(\lambda) \tag{3.23}$$

$$\mathbf{Var}(f) = -\frac{\partial^2}{\partial \lambda^2} \log Z(\lambda) \tag{3.24}$$

类似地，波耳兹曼-吉布斯分布P^*的熵可以表示为

$$\mathcal{H}\left(P^*\right) = -\sum_s P^*(s) \log P^*(s) = \log Z(\lambda) + \lambda \mathbf{E}(f) \tag{3.25}$$

统计力学的另一个中心问题是相变（phase transition）的研究，即研究当系统的某些参数——特别是温度T或等价的λ——发生变化时，系统行为发生的突变。如果$\mathbf{E}(f)$ 在λ_C处不连续，则称1阶相变发生在临界值λ_C处。如果$\mathbf{E}(f)$ 在λ_C处连续而$\mathbf{Var}(f)$ 不连续，则称λ_C处发生了2阶相变。相变的研究在学习理论中也十分重要，[252,482]但这已不是本书的研究范围。

3.2.3 自由能

由于分割函数的对数有重要作用［参见（3.23），（3.24）和（3.25）］，它也被称做自由能。更精确地，自由能$\mathcal{F}=\mathcal{F}(f, \lambda)=\mathcal{F}(\lambda)$ 定义为

$$\mathcal{F}(\lambda) = -\frac{1}{\lambda} \log Z(\lambda) \tag{3.26}$$

上述的式子显然可以改写为自由能的形式，例如

$$\mathcal{H}\left(P^*\right) = -\lambda \mathcal{F}(\lambda) + \lambda \mathbf{E}(f) \tag{3.27}$$

这个式子等价于

$$\mathcal{F}(\lambda) = \mathbf{E}(f) - \frac{1}{\lambda} \mathcal{H}\left(P^*\right) \tag{3.28}$$

上式有时也被看做自由能的另一种定义。在这个定义中，自由能依赖于函数f、参数λ和状态的分布P^*。因此该定义可以扩展到其他任意分布$Q(s)$：

$$\mathcal{F}(f, Q, \lambda) = \mathcal{F}(Q, \lambda) = \mathbf{E}_Q(f) - \frac{1}{\lambda} \mathcal{H}(Q) \tag{3.29}$$

其中$\mathbf{E}_Q$表示分布Q的期望值。这里我们没有考虑 f 的依赖性，实际上作为负对数概率的 f 的选择在统计应用中十分重要，例如下面和第4章中将要提到的EM算法的推导。将自由能与上述的拉格朗日算子比较，波耳兹曼-吉布斯分布明显等价于使得自由能达到最小的分布。

现在考虑任意两个分布$Q(s)$和$R(s)$，比较它们的自由能。一个简单的比较计算：

$$\mathcal{F}(Q,\lambda)-\mathcal{F}(R,\lambda)=\sum_s[Q(s)-R(s)]\left[f(s)+\frac{1}{\lambda}\log R(s)\right]+\frac{1}{\lambda}\mathcal{H}(Q,R) \tag{3.30}$$

其中$\mathcal{H}(Q,R)=\sum_s Q(s)\log(Q(s)/R(s))$是$Q$和$R$之间的相对熵。

注意一点，如果我们取s的能量为负对数似然度$f(s)=-\log R(s)$，其中R是某种状态的分布，那么波耳兹曼-吉布斯分布与$R^\lambda(s)$成比例。特别地，当$\lambda=1$时，系统的波耳兹曼-吉布斯分布就是R本身：$P^*(s,1)=R$，而且其自由能减少为0。进一步地，对于任意的其他分布Q，自由能之间的差别等同于相对熵

$$\mathcal{F}(Q,1)-\mathcal{F}(R,1)=\mathcal{H}(Q,R) \tag{3.31}$$

由于相对熵通常是非负的，因此$\mathcal{F}(Q,1)\geqslant\mathcal{F}(R,1)$，当且仅当$Q=R$时等号成立。另外，波耳兹曼-吉布斯分布可使自由能达到最小。还有一个需要注意的重要之处是，$\lambda=1$的情况并没有什么特别之处。例如我们可以定义$f(s)=-\log R(s)/\lambda$，从而得到$\mathcal{F}(Q,\lambda)-\mathcal{F}(R,\lambda)=\mathcal{H}(Q,R)/\lambda$。

3.2.4　隐变量情况

在许多建模情况中都存在隐/不可观测/潜变量或因素，记为H。如果D表示数据，我们假定隐变量和观测变量之间存在联合分布$\mathbf{P}(D,H|w)$，w是参数。在我们感兴趣的情况中，w通常表示模型的参数。从统计力学的观点来看，可以认为系统的状态是通过隐变量决定的。如果定义 f 为

$$f(H)=-\log\mathbf{P}(D,H|w) \tag{3.32}$$

那么，在$\lambda=1$时波耳兹曼-吉布斯分布由后验分布

$$P^*=P^*(H,1)=\mathbf{P}(H|D,w) \tag{3.33}$$

给出，自自能由

$$\mathcal{F}(P^*,1)=-\log\mathbf{P}(D|w) \tag{3.34}$$

给出，它是数据的负对数似然度。进一步地，对于任何其他分布Q，自由能之间的差别由

$$\mathcal{F}(Q,1)-\mathcal{F}\left(P^{*},1\right)=\mathcal{H}\left(Q,P^{*}\right) \tag{3.35}$$

或

$$\log\mathbf{P}\left(D|w\right)=-\mathcal{F}(Q,1)+\mathcal{H}\left(Q,P^{*}\right) \tag{3.36}$$

给出。在后验概率$\mathbf{P}(H|D,w)$及其相应的期望值很难计算时，为了使数据的似然度达最大，有时可以使用一些计算比较容易的次优策略，这些策略基于其他类型的分布Q，它们离真实的后验概率不会太远。关于最小化自由能$\mathcal{F}(Q,\lambda)$的讨论见参考文献［146, 255］以及附录A中有关变分法的章节。

第4章　机器学习算法

4.1　绪　论

在这一章里，我们将介绍在机器学习方法应用中涉及到的主要算法，这些算法将在本书的其余部分应用。我们将简要描述每一种算法，并为读者提供相关主题的大量参考文献。

我们已经看到，在根据一些数据构造出一个参数化的模型$M(w)$之后，接下来的任务是：

1. 估计联合分布$\mathbf{P}(w, D)$和后验概率$\mathbf{P}(w|D)$；
2. 估计参数w的最优解集，使得$\mathbf{P}(w|D)$最大，这是贝叶斯推断的第一层次；
3. 根据后验分布估计其边缘分布和期望值，例如计算形如$\mathbf{E}(f) = \int f(w)\mathbf{P}(w|D)\,dw$的积分，这是更高层次的贝叶斯推断。

因此，根据算法的目的是要估计概率密度、某个参数值，还是估计某个期望值，可将算法分为三类。虽然这种分类带有某种任意性，但实际需要仍要求我们采用这种分类。实际上，任何一个问题都可以变换为一个优化问题的形式，而某一事件发生的概率则是相应的指示函数（indicator function）的数学期望：$\mathbf{P}(A) = \mathbf{E}(I_A)$。与此类似，经常用于估计序列数据似然度的动态规划也可视为一种优化技术。

在4.2节中，我们将简要回顾动态规划算法——一种应用于序列分析的关键算法，以及它在序列似然度估计中的应用。在接下来的两节里，我们将考虑一些优化$\mathbf{P}(w|D)$的算法，包括梯度下降法和EM（期望最大化）/GEM（广义期望最大化）法。第4.5节讨论蒙特卡罗—马尔可夫链（Monte Carlo Markov chain method,

MCMC）方法在高维分布的随机采样及相关期望值计算中的应用。而模拟退火算法（simulated annealing）将推迟至4.6节讨论。这是因为模拟退火算法在很大程度上依赖于随机采样。在4.7节中，我们将简要介绍进化算法。在4.8节里，我们结合应用方面的问题进行一些总结和补充。

4.2 动态规划

若一个问题可以递归分解为两个规模较小的相似子问题，那么原始问题的解就可以通过这两个子问题的解合成而得。动态规划[66]正是解决这类问题的一种普适的优化技术。动态规划应用的原型是在一个图中寻求两节点的最短路径。显然，图中从节点A通过节点C到达节点B的最短路径就是由A到C的最短路径加上由C到B的最短路径，这称为“贝尔曼原则”（Bellman principle）。对这类问题的一般解法是通过递归地组合更短的最优路径进行构建。

动态规划和它的许多变体非常普遍地应用于序列分析中。Needleman-Wunch和Smith-Waterman算法，[401,481,492]以及其他序列比对算法（如电气工程师经常使用的Viterbi解码算法）都是动态规划的应用。序列比对算法可以被形象地视为在一个具有适当度量的图中寻找最短路径。对两个长度为N的序列进行比对意味着在一个有N^2个节点的图上找到一条最短路径。由于动态规划本质上要求对所有的节点遍历一次，所以它的时间复杂度是$O(N^2)$。

在第7章和第8章中，我们在训练和利用模型时将大量应用动态规划和Viterbi算法，以便计算似然度和利用HMM进行序列比对。因此，我们将在这里给出相应算法的详细起源，并对其他章节中所用到的动态规划的其他一些变形算法做简要概述或将它们留做习题。由于动态规划非常著名，是许多传统序列分析算法的基础，所以我们推荐读者阅读有关的大量文献（尤其是参考文献［550］和其中所提及的参考文献）。再励学习算法（reinforcement-learning algorithm）也是一类重要的学习算法，它可以被视为动态规划思想的一种一般化推广。[298]

4.3 梯度下降法

我们经常关心的参数估计问题是寻求最优模型$M(w)$，使得负对数后验概率$f(w)=-\log \mathbf{P}(w|D)$或负对数似然度$-\log \mathbf{P}(D|w)$最小。如果函数$f(w)$可微，就可以运用一种最古老的优化算法——梯度下降法——来寻找其极小值。正如它的名字所表示的，梯度下降法是一个迭代的过程。它可以表示成以下的向量

形式：

$$w^{t+1} = w^t - \eta \frac{\partial f}{\partial w^t} \tag{4.1}$$

这里，η表示步长大小，或者叫做学习率（learning rate），它可以是固定值，也可以在学习过程中加以调整。

由于常规的梯度下降法比较简单，在复杂参数模型中，根据梯度的实际计算方法不同，可以给出这一方法的不同实现方法。[26] 在图模型中，通常要求信息的反向传播。正如我们将要在下一章中看到的，这是梯度下降法应用于神经网络（反向传播算法）和隐马氏模型（前面—后向过程）的情形。很显然，梯度下降过程的结果依赖于初值估计。此外，如果需要优化的函数具有复杂的形式，那么一般情况下，梯度下降法会终止于局部极小值而不是全局最小值。因此如果可能，在应用时最好选取不同的起始点和学习率多次执行优化程序。

我们都知道，在某些情况下，简单的梯度下降往往是缓慢和低效的。为了克服这个问题，梯度下降法的许多变形应运而生。例如共轭梯度下降法，它运用2阶信息，或者由当前梯度和以前下降方向一起构成更复杂的下降方向。关于梯度下降法的更多细节和参考文献可以从参考文献［434］中找到。尽管梯度下降法较粗糙，但仍因其易操作性和实用性而被广泛采用。

4.3.1　随机方向下降法

有很多下降过程可以不必沿着梯度最大的路线进行。在梯度难于计算，计算所依赖的硬件的物理特性的直接支持以及逃离局部极小非常重要等情况下，随机方向下降法十分有用。例如，我们可以考虑在现有的估计上加一个随机扰动，并且只有当它低于现有水平时才接受它。否则采用相反的扰动，或者尝试一个新的扰动。在线性搜索算法中，下降的方向一旦决定，则沿着该方向找到最低点并接着产生另一个新方向。与线性搜索相关的思想以及随机方向下降法在下一节的EM算法和本章最后的进化算法中还会提到。

4.4　EM/GEM算法

另一类重要的优化算法是期望最大化（EM）和广义期望最大化（GEM）算法。[147,387] 这些算法有许多不用的应用，它们也应用于序列分析中。[352,113] 在HMM中，EM算法也被称为Baum–Welch算法。[54] 由于EM/GEM算法的用途不只限于HMM，我们将根据参考文献［400］的思路，利用第3章中提到的自由能的

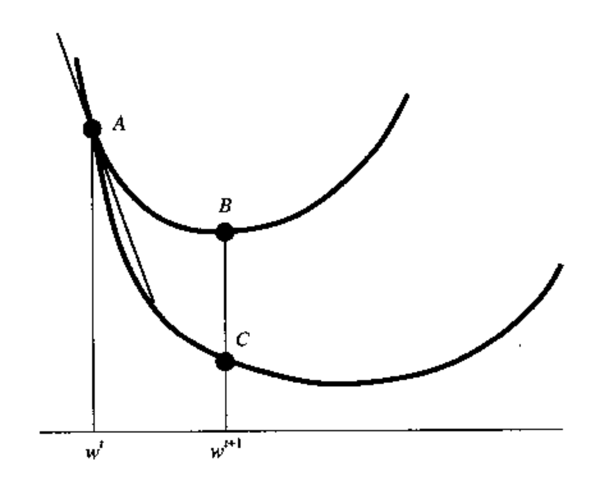


图4-1 EM算法的三个相继点

从w^t出发，为了使似然度曲面$F(w)=-\log\mathbf{P}(D|w)$达到最小，EM算法在约束条件$G(w^t)=F(w^t)=A$下最小化曲线$G(w)$。曲面$G$决定了曲面$F$，而且这两个曲面具有相同的梯度$w=w^t$。$w^{t+1}$处对应$G$的最小值点$B$，点$C$是通过计算隐变量的新的后验概率$\mathbf{P}(H|D, w^{t+1})$得到的。

概念，给出EM/GEM算法的一般处理方法。

在具有隐变量的模型和问题中，EM算法非常有用。典型的隐变量包括缺失或无法观测到的数据，混合模型中的混合参数和图模型中隐含的节点状态（如神经网络中的隐节点，隐马氏模型中的隐状态）。如果用D表示数据，我们假设有一个建立在隐变量和可观测变量基础上的参数化联合分布函数$\mathbf{P}(D, H|w)$，其中w表示模型的参数。以上处理方法的目标是最大化似然度$\log\mathbf{P}(D|w)$。这一思想同样可以应用于MAP估计中。由于直接优化$\log\mathbf{P}(D|w)$一般比较困难，我们的基本思想是试图优化它的期望值$\mathbf{E}(\log\mathbf{P}(D|w))$：

$$\mathbf{E}(\log\mathbf{P}(D|w))=\mathbf{E}(\log\mathbf{P}(D, H|w)-\log\mathbf{P}(H|D, w)) \tag{4.2}$$

EM算法是一个迭代算法，它交替执行两个步骤：步骤E（期望值）和步骤M（最大化）。在步骤E中，通过给出的观测数据和w的现有估计值，计算出隐变量的分布。在步骤M中，通过步骤E给出的隐变量的假定分布，计算出参数的最优可能值。设w在0时刻的初始估计值为w^0，在t时刻的EM算法可以更精确地写为如下形式：

1. 步骤E：计算H的分布$Q^*(H)$使得$Q^*(H)=\mathbf{P}(H|D, w^{t-1})$；
2. 步骤M：设置$w^t=\arg_w\max\mathbf{E}_{Q^*}[\log\mathbf{P}(D, H|w)]$。

正如在第3章中所看到的，如果我们定义一个隐含构型（hidden configuration）H的能量函数为$f(H)=-\log\mathbf{P}(D, H|w)$，则在$\lambda=1$时波耳兹曼-吉布斯分布由$\mathbf{P}(H|D, w)$给出。换句话说，EM算法的第一步就是使自由能关于$Q$最小：

$$\mathcal{F}(f, Q, 1)=\mathcal{F}(w, Q, 1)=\mathcal{F}(w, Q)=\mathbf{E}_Q(f)-\mathcal{H}(Q) \tag{4.3}$$

第二步就是使其关于f即w最小。这时，忽略常数$\lambda=1$，EM算法可以改写成如下形式：

1. 步骤E：计算波耳兹曼-吉布斯分布$Q^*(H)$使得$F(w^{t-1}, Q)$最小；
2. 步骤M：设置w^t使得$F(w^{t-1}, Q^*)$最小。

这里应该特别注意到，虽然Q^*依赖于w，但在步骤M中Q^*应保持不变。同样从第3章中还可以看出，波耳兹曼-吉布斯分布的自由能与数据的负对数似然度相同，即$\mathcal{F}(w, Q^*, 1)=-\log\mathbf{P}(D|w)$。

总而言之，EM算法就是一种通过在Q和w方向上进行交替优化，最终使得自由能$\mathcal{F}$达到优化的过程。这样就产生了如下形式的估计过程：

$$(w^t, Q^t)\rightarrow(w^t, Q^{t+1})\rightarrow(w^{t+1}, Q^{t+1})\rightarrow(w^{t+1}, Q^{t+2})\cdots \tag{4.4}$$

对于每一个t，它满足

1. $\mathcal{F}(w^t, Q^t)\geqslant\mathcal{F}(w^t, Q^{t+1})\geqslant\mathcal{F}(w^{t+1}, Q^{t+1})\geqslant\mathcal{F}(w^{t+1}, Q^{t+2})\geqslant\cdots$
2. $\mathcal{F}(w^t, Q^{t+1})=-\log\mathbf{P}(D|w^t)$
3. $Q^{t+1}=\mathbf{P}(H|D, w^t)$和$\mathcal{F}(w^t, Q^t)-\mathcal{F}(w^t, Q^{t+1})=\mathcal{H}(Q^t, Q^{t+1})$

通过上面的说明可以清楚地看到：除了一些极个别的鞍点，EM算法如我们所希望的那样最终收敛于$\mathcal{F}(w, Q)$的局部极小值点，这个值也是$-\log\mathbf{P}(D|M)$的局部极小值点。

单独从w的角度来看EM算法非常有启发性。假设在时刻t我们有一个估计值w^t，与其对应的似然度为$-\log\mathbf{P}(D|w^t)$。则

$$w^{t+1}=\arg_w\min\left[-\mathbf{E}_{Q^{t+1}}\log\mathbf{P}(H, D|w)\right] \tag{4.5}$$

其中$Q^{t+1}=\mathbf{P}(H|D, w^t)$。将$\mathbf{P}(H, D|w)=\mathbf{P}(H|D, w)\mathbf{P}(D|w)$带入并整理，上式等价于

$$w^{t+1}=\arg_w\min\left[-\log\mathbf{P}(D|w)+\mathcal{H}(Q^{t+1}, \mathbf{P}(H|D, w))\right] \tag{4.6}$$

这样，从w^t出发，通过EM算法可以找到曲面$G(w)=-\log\mathbf{P}(D|w)+\mathcal{H}(Q^{t+1},$

$\mathbf{P}(H|D,w)$）的最小值，它决定着我们想要优化的曲面$F(w)=-\log\mathbf{P}(D|w)$。这样整个优化过程就变为使似然度最大化来确保小的互熵，而不必离$\mathbf{P}(H|D,w^t)$的值太远。对G进行向量求导即得到

$$\frac{\partial G}{\partial w}=-\frac{\partial\log\mathbf{P}(D|w)}{\partial w}-\sum_H Q^{t+1}(H)\frac{\partial\mathbf{P}(H|D,w)/\partial w}{\mathbf{P}(H|D,w)} \tag{4.7}$$

当$w=w^t$时上式右边的第2项被消掉，这样：

$$\frac{\partial G}{\partial w}\bigg|_{w=w^t}=-\frac{\partial\log\mathbf{P}(D|w)}{w}\bigg|_{w=w^t} \tag{4.8}$$

新曲面G的切向量与原曲面$F(w)=-\log\mathbf{P}(D|w)$的切向量相同。因此，负对数似然度的梯度下降与EM算法的下降在同一个方向上（图4-1）。当分布$\mathbf{P}(D,H|w)$为指数族时，EM算法将变得更加简单。尤其是在这种情况下，函数G始终为凸函数。EM算法在分布为指数族的情况下的特殊性将留做习题。

最后，任何通过下降函数G（无需找到其最小值），进而改善似然度的算法被称为GEM（广义EM算法）算法。[147]前面的几何图形显示，在似然度上的梯度下降法可以看做是一种GEM算法（关于步骤E和步骤M如何分开执行以实现在线计算的讨论见参考文献［400］）。

4.5 马尔可夫链—蒙特卡罗方法

马尔可夫链—蒙特卡罗（MCMC）方法属于与统计物理有关的随机方法中重要的一类。现在它越来越多地被用于贝叶斯推断和机器学习中。[578,202,396,520,69]我们回忆一下，广义贝叶斯体系的一个基本目标是计算出高维概率分布$P(x_1,\cdots,x_n)$的期望值，其中x_i可以是模型的参数或者隐变量的值，也可以是观测到的数据。MCMC的两个基本思想非常简单。其中第一个思想（蒙特卡罗）是用下式估计期望值：

$$\mathbf{E}(f)=\sum_{x_1,\cdots,x_n}f(x_1,\cdots,x_n)P(x_1,\cdots,x_n)\approx\frac{1}{T}\sum_{t=0}^{T}f(x_1^t,\cdots,x_n^t) \tag{4.9}$$

对于较大的T，根据分布$P(x_1,\cdots,x_n)$进行采样得到（$x_1^t,\cdots,x_n^t$）。为了从分布P中采样，第二个思想即是构造一条马尔可夫链，使得它的平衡分布为P。然后对这条马尔可夫链进行模拟并试图对其平衡分布进行采样。

在我们考察马尔可夫链的基本原理之前，有几点值得注意。（4.9）右边估计的均值为$\mathbf{E}(f)$。如果采样点相互独立，它的方差即为$\mathbf{Var}(f)/T$。在这种情况下，估计值的精度便不依赖于采样空间的维数。重要性采样（importance sampling）和拒绝采样（rejection sampling）是产生独立采样点的两个著名的蒙特卡罗算法，这里我们对此将不做讨论。但这两个算法在高维状态空间中是低效的。用马尔可夫链方法产生的采样点是不独立的。但在平衡状态这些样本的分布为P。一个采样点对前一个样本的依赖性是在高维空间中MCMC方法具有较高效率的关键所在。毕竟，如果P可微甚至仅仅连续，一个样本的概率$P(x_1,\cdots,x_n)$能够同时提供它相邻样本的信息。甚至为了计算方便，P被视为一个归一化常数时，上述性质仍然成立。最后，MCMC方法与其他建立在单一点估计基础上的方法一样，最多只是对理想的贝叶斯推断过程的一种近似。它依赖于给定样本D计算出$\mathbf{P}(\mathbf{E}(f)|D)$的值。

4.5.1 马尔可夫链

马尔可夫链的理论基础已经十分完善。[176] 这里我们仅仅回顾一下最基本的概念，其他内容请读者参考有关的课本、文献。如同在统计力学中，考虑一个具有$|S|$个状态的系统$S=\{s_1, s_2, \cdots, s_{|S|}\}$。令变量序列$S^0, S^1, \cdots, S^t, \cdots$代表不同时刻的系统状态，这样从1到$|S|$的每一个整数分别代表这条链的一个状态。在任何时刻，这条链都处于一个特定状态。变量S^t构成一条马尔可夫链，当且仅当对于任何时刻t有：

$$\mathbf{P}(S^{t+1}|S^0,\cdots,S^t)=\mathbf{P}(S^{t+1}|S^t) \tag{4.10}$$

直观上我们还可以这样说：未来只通过现在与过去相联系。S^t称做t时刻链的状态。一条马尔可夫链可以由初始分布$P(S^0)$和转移概率$P^t=P(S^{t+1}|S^t)$完全确定。这里我们只考虑静态马尔可夫链，即转移概率是与时间无关的常数。这样马尔可夫链的转移矩阵可写为$T=(t_{ij})$，其中t_{ij}表示从状态s_j到状态s_i的转移概率。这里请注意，与（4.9）相对应，链的空间由坐标$x_1,\cdots,x_n$给出，也就是说每一个S^t是一个n维变量。

如果一条链的状态空间达到某种分布后就保持不变，则说此分布为平稳分布。因此平稳分布Q必须满足平衡方程

$$Q(s_i)=\sum_{k=1}^{|S|} t_{ik}Q(s_k)=\left(1-\sum_{j\neq i} t_{ji}\right)Q(s_i)+\sum_{j\neq i} t_{ij}Q(s_j) \tag{4.11}$$

或写成

$$-\sum_{j\neq i} t_{ji}Q(s_i)+\sum_{j\neq i} t_{ij}Q(s_j)=0 \tag{4.12}$$

这样，平稳的充分条件为下列平衡方程组成立：

$$t_{ji}Q(s_i)=t_{ij}Q(s_j) \tag{4.13}$$

其中i，j为表示状态的正整数。这表明由状态s_i到s_j平均转移次数等于由状态s_j到状态s_i平均转移次数，从而所有状态上的总体分布保持不变。

一条马尔可夫链一般有几个平稳分布。而有限状态空间的马尔可夫链至少存在一个平稳分布。显然，在MCMC采样过程中，实际上在要求更强的条件即各态遍历分布的情况下，我们主要关心平稳分布。这里我们定义一个分布为各态遍历分布，当且仅当不论这条链在0时刻的初始分布如何，它总会收敛到该分布。各态遍历马尔可夫链中只存在一个平稳分布，称为平衡分布（equilibrium distribution）。现在马尔可夫链具有各态遍历性的条件，以及它收敛到平衡状态的速率都已经很清楚了。[150,180]

为了达到从$P(x_1, \cdots, x_n)$中采样的目的，我们现在来讨论两个主要的MCMC算法：吉布斯采样（Gibbs Sampling）和Metropolis算法。

4.5.2 吉布斯采样

吉布斯采样，也叫热浴（heatbath）方法，是一种最简单的MCMC算法。[199] 它的适用范围很广，特别是当条件分布$P(x_i|x_j: j\neq i)$容易计算或变量x_i从一个很小的集合中取值时。在吉布斯采样过程中，依据所有其他变量当前的值，迭代地对其中每一个变量进行采样。从（$x_1^t, \cdots, x_n^t$）开始，

1. 依据$P(X_1| x_2^t, x_3^t, \cdots, x_n^t)$选取$x_1^{t+1}$的值；

2. 依据$P(X_2| x_1^{t+1}, x_2^t, \cdots, x_n^t)$选取$x_2^{t+1}$的值；

3. ……

n. 依据$P(X_n| x_1^{t+1}, x_2^{t+1}, \cdots, x_{n-1}^{t+1})$选取$x_n^{t+1}$的值。

据此，我们依次对变量进行循环操作。我们也可以采用其他循环顺序，或者在每一步中都选取相同的变量。甚至可以应用任何其他固定分布，只要每一个变量被访问的概率不为0。也可以对变量成组地采样，而不是一个一个地采样。由定义容易验证，吉布斯采样算法可以最终得到一个平稳分布。有关各态遍历性的证明和

其他内容可以在上述有关MCMC方法的一般文献及参考文献［209,191,490］中找到。附录C给了一个用于贝叶斯网络的吉布斯采样方程的例子。下面将要讨论另一个MCMC方法：Metropolis算法。吉布斯采样是这种算法的一种特例。

4.5.3 Metropolis算法

这里假定我们的目标仍然是从一个给定分布$P(s)=P(x_1,\cdots,x_n)$中采样。Metropolis算法[388]是对现有状态施加随机扰动，然后依据状态概率的变化来判断该状态是否可被接受。

准确地说，Metropolis算法中应用了两个辅助的分布族Q和R。$Q=(q_{ij})$为选择分布（selection distribution），其中q_{ij}表示处于状态s_j时选择状态s_i的概率。$R=(r_{ij})$为接受分布（acceptance distribution），其中r_{ij}表示处于状态s_j并选择s_i为下一个候选状态时接受状态s_i的概率。显然我们有$q_{ij}\geq 0$，$r_{ij}\geq 0$，且$\sum_i q_{ij}=1$。在本书下面的部分，并且在大多数实际情况中，可以假设Q具有对称性，即$q_{ij}=q_{ji}$，但这个假设并不是必要的。从t时刻的状态s_j开始（$s^t=s_j$），此算法的过程如下：

1. 根据分布q_{ij}随机选择一个状态s_i；
2. 以概率r_{ij}接受状态s_i，即$S^{t+1}=s_i$的概率为r_{ij}，$S^{t+1}=s_j$的概率为$1-r_{ij}$。

在Metropolis算法的最一般形式中，接受分布定义为：

$$r_{ij}=\min\left(1,\frac{P(s_i)}{P(s_j)}\right) \qquad (4.14)$$

我们把将“吉布斯采样方法可以写成Metropolis算法的形式”这一证明留做习题。当P被表示为能量函数的形式，即$P(s)=e^{-\mathcal{E}(s)/kT}/Z$时，（4.14）可写做

$$r_{ij}=\min\left(1,e^{-\left[\mathcal{E}(s_i)-\mathcal{E}(s_j)\right]/kT}\right)=\min\left(1,e^{-\Delta_{ij}\mathcal{E}/kT}\right) \qquad (4.15)$$

注意到这里只需要概率的比，而不需要分割函数本身。所以，此算法可以表示成我们更熟悉的形式：

1. 根据分布q_{ij}随机选择一个状态s_i；
2. 当$\mathcal{E}(s_i)\leq\mathcal{E}(s_j)$时，接受状态$s_i$；当$\mathcal{E}(s_i)>\mathcal{E}(s_j)$时，以概率$e^{-\Delta_{ij}\mathcal{E}/kT}$接受状态$s_i$；当状态$s_i$被拒绝时，停留在状态$s_j$。

可以很容易地验证，在Metropolis算法中分布P是平稳的。我们有$t_{ij}=q_{ij}P(s_i)/P(s_j)$和$t_{ji}=q_{ji}$。因此$Q$是对称的，立即可以得出

$$P(s_j)\,t_{ij}=P(s_i)\,t_{ji} \qquad (4.16)$$

换句话说，由于上述平衡方程组成立，所以P是平稳的。

保证各态遍历性的充要条件是确保链中不存在吸收态。也就是说从任意状态s_i到任意状态s_j总存在一条转移概率不为0的路径。当然，这取决于q_{ij}的结构。还有几点需要说明。当且仅当$q_{ij}>0$时，用一条边连接两点i和j，我们就可以构造一个图G。如果所得到的图是完全图（或者仅仅非常稠密），那么这条链显然是各态遍历的。这种Metropolis算法是全局的，因为当此图为稠密而非完全图时，从任意状态i转移到任意状态j都存在非0的一步转移概率或多步转移概率。当图比较稀疏时，则包含了多个局部的Metropolis算法。此时只要保证任意两点间至少存在一条通路，各态遍历性就依然成立。例如可以对各部分分别使用这一算法，每次扰动其中的一部分。在大多数的实际应用中，从点i到与之相邻的各个点j的选择概率q_{ij}是相同的。一般q_{ij}被定为0，但这并不影响上述所结果。

Metropolis算法还有几种变形和推广。例如，使用能量函数的导数、其他接受函数[242,396]或聚类蒙特卡罗算法。[510,547]在特定情况下，我们甚至可以去掉Q为对称的条件，只要对接受函数R做如下的修改，平衡仍然得以保持：

$$r_{ij}=\min\left(1,\frac{P(s_i)q_{ij}}{P(s_j)q_{ji}}\right) \tag{4.17}$$

4.6 模拟退火算法

模拟退火算法[321]（有关综述见参考文献［67］）是一种受统计力学启发的通用的优化算法。它将MCMC的思想，如Metropolis算法和降温过程相结合。它的名字起源于冶金学。在冶金学中，经过退火处理（缓慢冷却）的金属比经过淬火处理（快速冷却）的金属的性能要好。金属宏观的高强度对应于内部分子的低能量状态。

考虑函数$f(x_1,\cdots,x_n)$的最小化问题。不失一般性，我们假设对于任意点都有$f\geqslant 0$。一般我们可以认为f表示具有状态$s=(x_1,\cdots,x_n)$的一个统计力学系统的能量。我们已经知道，在温度T时，系统处于状态s的概率可由波耳兹曼-吉布斯分布给出：$P(s)=P(x_1,\cdots,x_n)=e^{f(s)/kT}/Z$。理解模拟退火算法的首要关键是：在低温下，波耳兹曼-吉布斯分布主要被能量最低的状态占据，即它们成为最可能的状态。实际上，如果函数f达到最小值时的状态为m，我们有

$$\lim_{T\to 0}P(s)=\begin{cases}1/m & \text{如果}s\text{为能量基态}\\ 0 & \text{其他情况}\end{cases} \tag{4.18}$$

如果我们能够在0度附近模拟系统，我们将立即找到能量基态，即f的最小值。问题是在一般情况下，任何MCMC方法都不能在有限时间内达到波耳兹曼-吉布斯平衡分布，因为在状态空间中的移动受到一些概率极小的区域（高能势垒）的限制。模拟退火算法试图通过起始于波耳兹曼-吉布斯分布接近于平均分布的高温状态，并根据逐步降温的退火过程来解决这一问题。由于模拟退火经常与Metropolis算法结合使用，它实际上可以适用于任何MCMC方法，特别是吉布斯采样。

退火过程具有至关重要的作用。有许多理论结果[199]表明，退火过程满足下列对数形式：

$$T^t = \frac{K}{\log t} \tag{4.19}$$

其中$t \geqslant 1$。对于常数K的某些取值，这一算法几乎一定收敛于某一基态（有关K的下限见参考文献［230］）。（通过上下文区分表示温度的T和表示时间的T。）直观上，这一点很容易理解。[396] 如果我们用$s_{\max}$和$s_{\min}$分别表示具有最大能量和最小能量的两个状态，从波耳兹曼-吉布斯分布可以得出

$$\frac{P^t(s_{\max})}{P^t(s_{\min})} = \left(\frac{1}{t}\right)^{\Delta\mathcal{E}/kK} \tag{4.20}$$

其中$\Delta\mathcal{E}=\mathcal{E}(s_{\max})-\mathcal{E}(s_{\min})$。如果取$K=\Delta\mathcal{E}/k$，我们就有$P^t(s_{\max})=P^t(s_{\min})/t$。这样，对于任意状态$s$，有

$$P^t(s) \geqslant P^t(s_{\max}) = \frac{1}{t}P^t(s_{\min}) \geqslant \frac{1}{t}P^1(s_{\min}) \tag{4.21}$$

特别注意一点，在退火过程中，任一状态s被访问次数的下限为$P^1(s_{\min})\sum_t 1/t$，而$P^1(s_{\min})\sum_t 1/t$是发散的。这样，当K的取值对应于最高能量势垒时，此算法将不可能再收敛于局部极小值点。

但是必须注意到，对数形式的退火过程非常缓慢，一般无法应用于实际计算。它要求访问绝大部分可能的状态，这就几乎等同于穷举搜索。因此采用这种方法必然会得到全局最优值。另一方面，如果有一种方法可以替代穷举搜索，它必然会优先得到应用。我们关心的大部分问题都是NP完全问题，它们具有指数级的可能状态，使得我们无法应用穷举搜索法。在实际中，模拟退火必须采用更快的形式，例如采用几何退火的形式：

$$T^t = \mu T^{t-1} \tag{4.22}$$

其中$0<\mu<1$。自然地，在这种情况下，所能期望得到的最好结果是普遍地收敛到对应于低能点的近似解，而不是全局最小值。

其他与模拟退火算法[547,381]和MCMC基本思想相关的算法，如动态混合式蒙特卡罗方法，[152,396]在参考文献中有详细讨论。

4.7 进化和遗传算法

进化算法[261,476]（evolutionary algorithms）具有特殊的意义，因为它的启示来源于进化，而进化正是我们研究的核心领域。进化算法是优化算法中的一个大类，它试图通过某种方式模拟我们自认为了解的进化过程的内在机制。这一类算法的共同组成部分是产生随机扰动或突变，通过提供一个适应函数来对所给点进行评估，并滤除那些不适用的突变。在这个意义上，随机下降法甚至模拟退火算法都可以看做是一种特殊的进化算法。进化算法的一个最大子类是遗传算法（genetic algorithms）。

遗传算法[328,330]和相关的人工生命领域通过模拟点群在适应度空间的进化，将模拟进化又向前推进了一步。而且，除了利用突变，遗传算法还通过大量模拟基因操作和有性繁殖的其他方式（如交叉）产生新的点。虽然遗传算法非常灵活，并使复杂的事物如计算机程序的进化成为可能，但是这种算法在现有的计算机上运行得很慢。遗传算法在分子生物学上的应用见参考文献［329,233,415］。在参考文献［53］及其所提到的参考文献中还介绍了其他进化算法。本书将不再对它们做进一步的讨论。

4.8 学习算法的相关技术细节

与学习算法相关的许多实现细节、启发法和技巧是十分重要的。有关这些技巧的大量材料可以在NIPS（神经信息处理会议）的年会论文集中找到。这里我们只是从一般的角度讨论其中的部分问题。一些与特定模型有关的技巧将在相关章节中给出。

4.8.1 模型复杂度控制

在某种方式上，建模者总是要面对一个问题，即在数据的欠拟合与过拟合之间、在模型自由度的高与低之间寻求平衡。这个问题的解决方案之一，是在真实的似然函数中加入代表模型复杂度的一项，用于正则化。这一方法的基本原理基

于一些将训练误差$\mathcal{E}_T$和推广误差$\mathcal{E}_G$关联起来的等式和约束条件。约束条件一般为$\mathcal{E}_G \leqslant \mathcal{E}_T + C$，其中$C$反映了模型的复杂度。这个公式的例子可以在参考文献［533］和参考文献［5,16］中找到，前者应用了VC维数的概念，而后者应用了统计逼近理论。这样，通过最小化正则化的训练误差$\mathcal{E}_T+C$来最小化推广误差$\mathcal{E}_G$。其中$\mathcal{E}_T$表示数据拟合度，C则经常被视为偏好简单模型的先验知识。这样的处理方法可以得到很好的结果并具有启发价值。但是正如第2章指出的，从贝叶斯分析的角度来看，这种方法也存在一些弱点。对于复杂的数据，期望数据是由简单模型产生的这种先验知识没有意义。一般来说，我们建议采用更加有效的，自由度很高的模型，并通过赋予模型的参数和结构更大的自由度和较强的先验概率控制过拟合问题，而不是直接限制整个模型的复杂度。

4.8.2　在线/成批学习

在数据到来之时或在每个样本提交后，就进行一定的模型拟合和参数调整，这种学习方式被称为在线方式（online learning）。另一方面，如果参数值只是在大量样本（如果不是整个训练集的话）提交后才进行调整，这种学习方式被称为成批（batch）或离线（offline）方式。很显然两者之间存在着一系列可能的方式。在线学习在一些方面存在优势，它不需要记忆很多训练样本，这使它更具灵活性和易于应用。另外它可以随着数据的到来更新自己的信念（belief），这更接近贝叶斯分析的精髓，而这似乎正是生物系统学习的方式。更重要的是，随着每个样本的到来而进行的学习可以引入一定程度的随机性，这样有助于搜索解空间，避免局限于某个局部极小值点。当然还可以证明，当学习率充分小时，在线学习即可近似为成批学习（见参考文献［49］）。正因如此，在这本书中我们一般只提供在线学习方程。

4.8.3　训练/测试/检验

一种广泛应用的方法是只用数据的一部分进行模型拟合，而用剩下的数据或其中一部分进行模型的检验。应该看到这样的应用方法不完全是贝叶斯分析的方法，因为在第2章所提到的一般体系中，所有数据都被用于模型拟合，而无需进行检验。实际上，交叉验证技术仍非常有用，因为它们一般容易应用并能产生较好的结果，特别是当数据总数充分的时候。另一点要说明的是，可以有许多方法将数据分割为不同的子集并指定它们用于训练还是检验。例如，不同的数据集可以训练不同的专家模型，然后再将它们合并起来，而检验集可以用来决定超参数的值。当数据相对有限时，这种处理方法就变得更加重要。因此只要可能，最好

拥有三个不同的数据集：一个用于训练，一个用于检验和训练调节，一个用于总体性能的测试。

在生物信息学中，序列很可能由于具有共同的祖先而相互关联，因此还有一些需要特别注意的地方。在第1章中详细论述了构造低相似性测试集的问题，这对于可靠评估机器学习方法的预测性能可能具有重要意义。

4.8.4 提前结束

当一个模型相对于可用数据过于灵活（因为包含了过多的参数）时，在训练过程中将观察到过拟合。这意味着，当训练误差随着训练次数单调递减时，检验集上的误差起初也跟着减小，但其到达某一最小值时又开始增加。这样，过拟合与模型对训练数据的记忆或数据拟合中的噪声相联系，达到了一种不利于推广的程度。这种情况下，当然得对模型进行修改。另一种广泛应用的折中方法是提前结束（early stopping），即当训练误差率达到某一阈值或完成一定次数的训练周期后停止训练，但阈值或者周期次数不容易确定。一种可行的方法是，当误差率在不同于训练集的检验集上开始上升时即停止训练。这种方法的缺点在于必须牺牲一部分数据用于检验。并且，这种形式的提前结束仍然会导致测试数据中用于检验的数据的部分过拟合。换句话说，用于决定何时停止训练的模型在检验集上的性能，一般要优于它在新数据上的总体推广性能。即使这样，同其他检验方法一样，提前结束方法在实践中易于应用，特别是对于数据充足的情况。

4.8.5 模型集/系综

当一个复杂模型采用ML或MAP优化方法来拟合数据时，可以通过在学习过程中改变许多条件，如初始参数值、学习率、样本的提供顺序、训练集等，来得到不同的模型参数。另外，还可以试验不同类型的模型。通过对不同模型或专家的看法进行某种平均，可能得到更好的预测或分类（附录A和参考文献［223,237,277,568,426,340,339］）。解决一个特定问题的模型集（ensemble）又称为一个系综，这与统计力学相似（参见文献中决策机制的相关概念）。这一直觉看法的数学基础是，对于凸的误差函数，系综的整体误差小于每个成员的平均误差（附录B：Jensen不等式）。这样，总体的性能要优于某一个专家成员。有多种方法用于组合由不同模型产生的预测。广泛采用的方式是进行平均，也可以有其他方法，例如加权平均，包括训练时学习权重的可能性。注意对于在第2章的贝叶斯体系中完全确定的那类模型，最优预测是通过在所有可能模型上的积分得到的［参见（2.18）］。因此，模型的平均即可以作为这种积分的近似。

4.8.6　平衡和加权方案

一个需要考虑的重要问题是训练集是否平衡。在二项分类问题中，可用的正样本数量与负样本数量可能相差很大。同样地，在多项分类中，每一类可得到的数据的比例也会存在很大差别。这种情况在生物数据库中更加严重。例如在第1章中所描述的，由于大量的不同因素，会造成某些生物体或某些类别的序列高显现（overrepresented）。

理想情况下，为了正确分类，所有相关类的数据在训练集中出现的机会应该相等。第6章将描述这种平衡训练的方法。在一些情况下，训练集中某个类的数据显现度低将导致这个类的预测性能很差。这种情况经常作为信息缺失的依据，例如：进行β折叠预测就需要比进行螺旋预测更多的长程序列信息。虽然任何蛋白质结构预测方法都可以通过加入适当的长程信息而改善，但通过应用平衡训练方案，β折叠的性能就可以得到显著提高。[452]

另一种解决方法是使用加权方案人工平衡训练集。它等效于将稀少的样本进行多次复制。为了研究DNA和蛋白质序列，尤其是多重序列比对问题，人们设计了一系列加权方案。[10,536,487,201,249,337] 参考文献［337］中的加权方案非常有趣，它从最大熵的角度看是最优的。

由于篇幅关系还有许多技术我们没有涉及到。它们都可以在NIPS的年会论文集和其他有关神经网络技术的文献中找到。这些技术包括：

- 主动采样。
- 剪枝方法。这是一种在学习过程中或过程后对模型进行简化的方法。一般来说，它包括寻找方法来决定模型中哪些参数对模型的性能几乎没有影响，然后删除它们。多余的参数不光是指那些数值很小的参数，还包括那些互相抑制而对模型贡献很小的参数。
- 2阶信息估计方法。这些方法通过计算或估计似然度的Hessian矩阵，从而利用2阶信息调节学习率或计算误差带等。对Hessian矩阵的有效估计是一个有趣的问题，必须针对具体的模型加以考虑。

第5章　神经网络：理论

5.1 概　述

人工神经网络（artificial neural network），也称神经网络（NN），[456,252,70]它的提出源于模拟大脑的信息处理和学习过程。虽然模仿大脑依然是建模的灵感源泉，但现在大多数人工神经网络中使用的神经元与生物神经元有很大的差别。[85]人工神经网络不断发展，它在各个领域（包括计算分子生物学领域）都获得大量的实际应用。人工神经网络技术日益成为解决序列分析和模式识别问题的机器学习技术的一项重要工具。

根本上而言，神经网络可以看做参数化图模型的一大类别，这些图模型是由一些随时间变化、交互连接的神经元组成的网络。本书中仅使用点到点的相互连接构成神经网络，当然如果需要，可以使用与多个神经元节点相关的复杂连接，来构造高阶或“sigma-pi”类型的神经网络。[456]由神经元j到神经元i的连接权重记做w_{ij}。这样，神经网络可表示为带权重的有向图或结构。为了简化起见，我们不考虑节点的自连接，即假设对所有神经元都有$w_{ii}=0$。

首先需要了解一些重要的神经网络结构类型，如反馈（recurrent）、前馈（feed-forward）和分层（layered）结构。反馈网络是一种含有有向环的网络。不含有向环的网络称为前馈网络。由于反馈结构的网络动态特性较复杂，将在第9章中专门讲述。如果神经元被分为几层，而且各层之间存在连接，该类网络为分层网络。前馈网络不一定是分层网络。

本章中涉及较多的，目前在分子生物学中应用较多的网络结构为分层前馈网络，如图5-1所示。神经元节点通常分为可见节点（visible unit）和隐节点（hidden

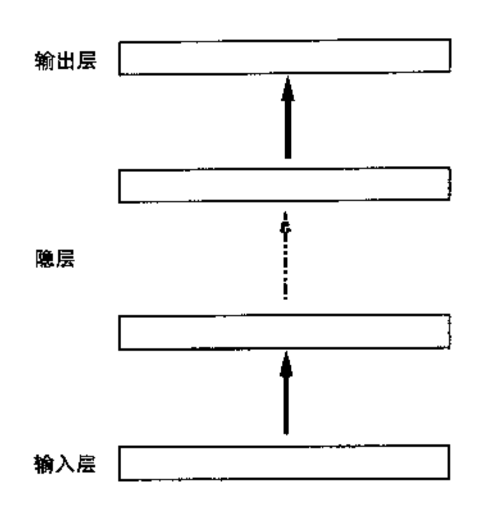


图5–1分层前馈网络结构/多层感知器（MLP）

各层包含数目不等的神经元，层之间的连接方式也各异。

unit）两类。可见节点指直接与外界作用的神经元节点，如输入、输出神经元节点。大部分情况下，在简单网络中，输入、输出神经元组成层结构，形成输出层和输入层。只包含隐节点的层称为隐层。神经网络的规模常常以层数衡量。当然，可以简单神经网络的模块或层次模式进一步构造更为复杂的整体网络结构。神经网络可见层的设计取决于用于序列数据编码的输入方式，以及通常代表结构与功能特征的输出方式。

每个神经元节点的动态行为可以用微分方程或离散差分方程（见参考文献［26］）描述。本书仅涉及离散差分方程形式。在分层前馈神经网络中，同一层中所有的神经元节点同时进行更新，而各层逐次顺序更新。有时采用随机型神经元节点效果更好（参见附录C关于图模型和贝叶斯网络的部分）。本章中将重点讨论确定型的神经元节点。通常节点i获得与其连接的所有节点的输入的总量，记做x_i，产生输出$y_i=f_i(x_i)$，其中f_i是该节点的激活函数（transfer function）。一般地，同一层的所有节点具有相同的激活函数，总的输入量为前一层节点总输出量的加权和。节点i的输入、输出量如下所示：

$$x_i = \sum_{j \in N^-(i)} w_{ij} y_j + w_i \tag{5.1}$$

$$y_i = f_i(x_i) = f_i\left(\sum_{j \in N^-(i)} w_{ij} y_j + w_i \right) \tag{5.2}$$

其中w_i为节点的阈值。这也可以看做加入一个连接权重为w_i、输出恒为1的附加节点。权重w_{ij}和w_i为神经网络的参数。在更一般的神经网络中，还可能拥有其他参数，例如时间常数、增益、延迟等。在本文涉及的网络中，参数的总数由网络层数、每层的节点数、层之间的连接方式决定。层之间的一种标准连接模式为“全连接”，即前一层的每个节点与下一层的每个节点都相连。局部连接模式越多，网络结构越经济。然而值得注意的是，与所有节点的全连接相比，层之间的连接数目，即使是“全连接”方式的连接数目，也是稀疏的。在参量恒值传递的情况下，对于前一层中的一组节点的输出，层中的每个节点操作相同。这样，一种简单连接模式可以在给定层中共享。在神经网络中，这被称为“权重共享”。这种技术普遍应用于图像处理问题中，在使用不同距离测量来区别特征的序列分析问题中，它也获得了成功。这种权重共享的方法定义了一个卷积核（滤波器），对输入做统一处理。使用权重共享，即使各层的节点数目较多，关联两层的自由参数的数量也会较少。在6.3节的二级结构预测中，举出了该技术应用的一个实例。

很多形式的激活函数被广泛使用。例如，在回归问题中，激活函数是线性的（如恒等函数），此时的节点称为线性节点。然而更多的时候，激活函数是非线性的。有界激活函数也被称为压缩函数（squashing function）。当f为一个阈值函数时，

$$f(x)=\begin{cases}1 & 若x>0\\ 0 & 其他情况\end{cases} \tag{5.3}$$

此节点也称为阈值门节点（threshold gate unit）。阈值门节点实现二值决策功能，该决策以对相关节点进行加权评价为基础。显然，阈值的选择决定了阈值界线的位置。本书统一采用（0, +1）的值域，等同于如（-1, +1）之类的其他值域定义范围。阈值门节点是不连续的，所以常常用sigmoidal激活函数取代，这种取代的优点在于使函数连续可微。本书中，采用logistic激活函数

$$f(x)=\sigma(x)=\frac{1}{1+e^{-x}} \tag{5.4}$$

专门用于估计二值随机事件的发生概率。而使用诸如$f(x)=\tanh(x)$和$f(x)=\arctan(x)$之类的sigmoidal激活函数可以取得相同的结果。必要的情况下，可以为每个神经元节点引入增益λ_i，此时激活函数记做$y_i=f_i(\lambda_i x_i)$。另一种重要的神经元节点类型为“归一化指数节点”，也称做softmax函数，常用于计算具有n个可能输出的事件的发生概率，例如n个类别的分类问题。下标j从1到n，代表n个输出节点。计算出n个成员的概率，x_j表示第i个输出节点的输入总量，则每个输出节点

的最终输出量y_i为

$$y_i = \frac{e^{-x_i}}{\sum_{k=1}^{n} e^{-x_k}} \tag{5.5}$$

显然，其中 $\sum_{i=1}^{n} y_i=1$。当$n=2$时，归一化指数函数可通过简单变换表示成logistic函数形式：

$$y_1 = \frac{e^{-x_1}}{e^{-x_1}+e^{-x_2}} = \frac{1}{1+e^{-(x_2-x_1)}} \tag{5.6}$$

值得注意的是，任何一种概率分布$P=(p_i)$（$1 \leqslant i \leqslant n$）都对应一组变量$x_j$（$1 \leqslant j \leqslant m$），并可用以下归一化指数函数的形式表示：

$$P_i = \frac{e^{-x_i}}{\sum_{k=1}^{m} e^{-x_k}} \tag{5.7}$$

其中$m \geqslant n$。令$x_i=\log p_i+K$（$i=1,\cdots,n$）（如果必要，可令$j>n$时$x_j=-\infty$），其中K取某一正常数，故p的表示形式不惟一。而$m<n$时，该式无确定解，除非假设p_i至多可取m个不同值。

另一类广泛使用的激活函数为径向基函数（radial basis function，RBF），其中f一般为钟型函数（如高斯函数）。每个RBF节点i有一个“参考”输入 x_i^*，f是输入与“参考”输入的距离$d(x_i^*, x_i)$的函数，函数中的距离由节点输出$y_i=f(d(x_i^*, x_i))$计算。在空间问题中，d一般是欧几里德距离。

显然，建模者应该能够根据待解决问题的需要，设计选择合适的节点类型、连接和激活函数。所以在读者印象里，神经网络的概念往往是模棱两可的，而实际上也确实如此。按照我们给出的宽松定义，将多项式认做一种神经网络也未尝不可。当然，也可以进一步限制神经网络的定义范畴。传统意义上的神经网络特指输入满足（5.1），激活函数为阈值函数或sigmoidal函数的网络结构。实际上，具体的网络形式是随着具体的问题而定的。今天使用的不同类型网络模型的术语，部分地是历史偶然的产物。实际上神经网络模型是一族可能的参数化模型，是一个连续序列，尚没有明确的范围限定。设计网络结构和进行贝叶斯推断时，建模者的自由度很大。

在神经网络的实际应用中，往往需要对回归和分类识别这两类问题加以区分。在回归问题中，其目的是逼近或拟合给定曲面；而在分类识别问题中，其目的是

将给定输入划分入数量较少的几类中。这种区分虽然有用，但往往较为随意，比如，两类别的分类问题可以看做是对一个非连续的二值函数（有界的）的拟合。第6章中提及的遗传密码学习问题就是这两类问题重合的一个实例。由于序列数据的离散特性和存在识别一些典型特定模式（例如α螺旋、折叠类、剪切位点、外显子等）的问题，神经网络的分类功能过去在分子生物学中应用较多。但不可忽视的是，诸如疏水性标度（hydrophobicity scale）和堆积能（stacking energy）等连续数据也是很重要的。在以下的章节中，将对神经网络的回归、分类功能做更详细的介绍。

神经网络的最重要的特性之一就是可以通过样本进行学习。显然，从一般的贝叶斯统计理论体系的角度看，所研究的问题无非是模型拟合和参数估计。需要进行回归和分类的数据D通常是以输入—输出二元组的形式出现的：$D=(D_1,\cdots,D_K)$，其中$D_i=(d_i, t_i)$（d为数据输入，t为目标）。实际应用时，数据通常被划分为训练数据（training data）和校验数据（validation data）。训练数据用于模型拟合，校验数据用于模型检验。校验数据也可分为校验数据（validation）和测试数据（test data）两类，其中校验数据用于执行提前结束以避免神经网络的过学习，而测试数据用于评价模型的整体性能。这种输入数据与相应输出目标值都已知的模型拟合，常被称为有监督学习（supervised learning）；而相应输出目标值未知的模型拟合，则称为无监督学习（unsupervised learning）或自组织（self-organization）。当然，这种人为的划分有一定意义但不必作为教条。对于有监督学习算法，一种过去常用的思想是：从一组随机参数开始定义一个“误差函数”，这个误差函数是通过比较网络的实际输出和目标输出的差别而得到的。然后采用梯度下降法，修正优化网络参数值，使误差函数值最小。上述过程能用一般的贝叶斯统计理论（参考第2章）加以最完美地分析，即：先建立一定的概率模型和假设，然后进行合适的贝叶斯推导。神经网络有监督和无监督学习的许多算法，其实质就是ML或MAP估计。

本章的其余部分将集中讨论分层前馈神经网络结构、多层感知器［输入如（5.1），激活函数为线性/阈值/sigmoidal/归一化指数形式］及它们在序列分析中的应用。在下一节中，将详细分析神经网络的通用函数逼近特性。尤其是将证明存在一个足够大的、层数有限的神经网络，它能以任意精度逼近任何满足一定条件的函数。在5.3节中，将采用第2章中所述的理论框架分析神经网络、先验分布和似然函数，阐述如何设计神经网络结构，如何实现第一级的贝叶斯推断。在5.4节中，将采用第4章所介绍的理论来分析神经网络的学习算法问题，从而导出著名的反向传播学习算法（backpropagation algorithms）。神经网络的其他理论结果超出

本书范围，读者可参考所附文献。神经网络的计算复杂度问题和一般的机器学习问题见参考文献［314］。参考文献［373, 398, 517］给出了更完整的神经网络的贝叶斯方法，其中包括高级的贝叶斯推断方法。除了神经网络，还有一些其他适用于回归、分类的参数化模型，如样条函数[546]、高斯过程[559,206,399]（附录A）和支持向量机。[533,475]

5.2 通用函数逼近特性

神经网络的另一大特性就是能够以任意精度逼近任意给定函数。由于任意布尔函数都可以由多个阈值门的组合实现，所以对布尔函数而言，以上结论显然成立。③ 布尔函数可以由“与门”和“非门”构成，而“与门”和“非门”很容易由阈值门实现。在一般的回归问题中，只要隐层的节点足够多，任何实函数$f(x)$都可以由一个输入层为x，隐层为sigmoidal节点，输出层为线性节点的三层神经网络做任意精度的逼近。这一结论的多种数学推导和证明方法见参考文献［264, 265］等。

这里仅针对一种特殊情况给出简单的构造性证明，用以阐述其中的某些基本思想，而该证明能很容易地加以推广。为了简单起见，仅考虑连续函数$y=f(x)$，其中x和y仅为一维的情况。不失普遍性，假设x的取值范围为［0, 1］，计算任意给定x时$f(x)$的值，并满足精度ε。由于f在紧集［0, 1］上是连续的，所以f也是一致连续的，且存在整数n，使得下式成立：

$$|x_2 - x_1| \leqslant \frac{1}{n} \Rightarrow |f(x_2) - f(x_1)| \leqslant \varepsilon \tag{5.8}$$

因而存在一个函数g可以充分逼近f，其中$g(x)$满足：$g(0)=f(0)$，$g(x)=f(k/n)$（x的取值范围为$((k-1)/n, k/n]$，$k=1,\cdots,n$）。函数g可由以下类型的神经网络实现：仅有1个输入节点x，隐层为$n+1$个阈值门型节点且每个隐层节点皆与输入节点相连，输出节点也仅为1个，并与每个隐层节点连接。隐层节点编号为0至n，输出为线性激活函数，以覆盖y的取值范围（如图5-2所示）。输入节点到所有隐层节点的权重都设为1，第k个隐层节点的阈值（偏移）为$(k-1)/n$。此时，任取$x\in((k-1)/n, k/n]$，除前$k+1$个隐层节点的输出为1外，其余隐层节点的输出皆为0。这样，输入值直接由激活的隐层节点数编码。从第k个隐层节点到输出

③ 本节中所指节点一般为阈值/sigmoidal节点，显然结论对于包含多项式节点的神经网络同样成立。

节点的连接权重为$\Delta_k f=f(k/n)-f[(k-1)/n]$，其中$\Delta_0 f=f(0)$。输出节点的激活函数类型为零偏倚的恒等函数，因此若x=0，则$g(x)$=0。对于任何k=1, 2,⋯, n，若$x\in[(k-1)/n, k/n]$，则$g(x)=f(0)+\sum_{j=1}^{k}f(j/n)-f[(j-1)/n]=f(k/n)$。

利用上面的结论，很容易做出如下推广：

1. 多维输入、输出；
2. 激活函数或其他类型的激活函数；
3. 输入可在任意紧集区间上取值；
4. 函数f可以具有有限数量的不连续点，甚至更多。

神经网络可以逼近任意函数这个结论虽然有用，但上述证明并未给出一个非常实用的网络结构。实际上，可以证明对于基本的随机的函数，逼近函数的简单网络结构并不存在。仅对一类特定的“结构化”的函数，才存在简单的逼近网络，而此时通用函数逼近理论所构造的网络结构远非最佳。可能存在更优的网络结构，它更合理地分配隐节点，并可能具有不止一个隐层。正是这些因素，才使神经网络的学习算法尤为重要。

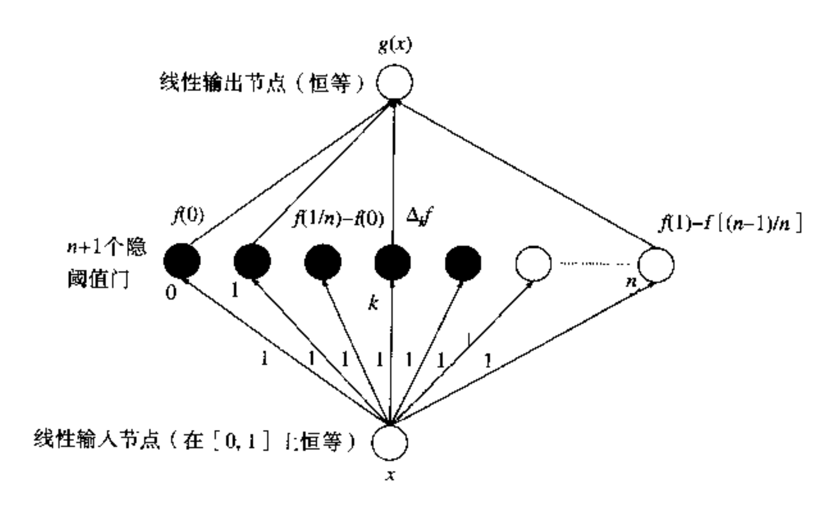


图5-2　函数的通用逼近网络结构

函数$f(x)$的逼近函数$g(x)$由以下神经网络实现：带有1个输入节点，n+1个阈值门型隐层节点和1个线性的输出节点。

5.3　先验分布和似然度

下面将应用第2章中所讲述的理论。特别地，将示范如何在理论的指导下选

择目标函数和输出层节点的激活函数。在本节中，假设数据由一组互相独立的输入—输出二元组$D_i=(d_i, t_i)$组成。由于噪声的影响，对于给定输入d_i，可以观测到不同的输出t_i。附加于输入d的噪声可以通过建模描述，但这里暂不予考虑。而神经网络本身的运算可看做是确定的。从而：

$$\mathbf{P}((d_i, t_i)|w) = \mathbf{P}(d_i|w)\mathbf{P}(t_i|d_i, w) = \mathbf{P}(d_i)\mathbf{P}(t_i|d_i, w) \tag{5.9}$$

后一个等式成立的条件是：通常假设输入d与参数w独立。因此，对于参数为w的给定结构的神经网络，根据（2.9）可推出下式：

$$-\log\mathbf{P}(w|D) = -\sum_{i=1}^{K}\log\mathbf{P}(t_i|d_i, w) - \sum_{i=1}^{K}\log\mathbf{P}(d_i) - \log\mathbf{P}(w) + \log\mathbf{P}(D) \tag{5.10}$$

推导中使用了$\mathbf{P}((d_i, t_i)|w)=\mathbf{P}(d_i)\mathbf{P}(t_i|d_i, w)$的结论，并考虑了不同数据点的独立性。依照第一级的贝叶斯推断（MAP规则），我们希望等式左边最小。而$\mathbf{P}(D)$、$\mathbf{P}(d_i)$与w无关，所以可暂不考虑，从而集中考虑似然项和先验项的大小。

为了计算似然度，需要区分问题的类别（如回归、分类），并进一步给出确定的概率模型。其具体步骤见参考文献［455］。其基本思想是：对于给定输入d_i，网络可以产生输出$y(d_i)$。当从统计意义上确定了如何由网络输出$y_i=y(d_i)$获得观测数据$t_i=t(d_i)$时，整个模型基本上就确定了。若输出层有多个节点，记y_{ij}为第i个样本在第j个节点上的输出。为了注释方便，下面将去掉输入的下标。如此可以获得针对一般化的输入—输出二元数据组（d, t）的在线计算等式。而离线计算等式可以很容易地依照（5.10）和将输入相加获得。

5.3.1 先验分布

如果不考虑其他附加信息，最自然并最广泛使用的神经网络参数的先验分布遵循零均值的高斯分布。连接权重、偏移和不同层的节点可选择不同的超参数（如高斯分布的标准差）。若权重w取值满足标准差为σ的高斯先验分布，则对应负对数后验概率相关计算项的值为一常数：$w^2/2\sigma^2$。这一常数也可以看做是一个正则因子，用以惩罚常常导致过拟合的大权重值。在权重的梯度下降学习算法中，权重w的更新方程中就包含一项因子$-w/\sigma^2$。这一因子也称为权重衰减因子（weight decay）。权重共享是另一种特殊的先验分布，此时，给定层中的不同节点组具有相同的输入连接权重。在梯度下降学习中，可以很容易地实现权重共享。权重共享在一类具有平移不变性特征，需要完成一些相同操作的问题（比如在输入的不同位置区域中提取模式特征）中是很有用的。这些共享的权重实质上完成

了卷积核的功能，所以这类网络被称为卷积网络（convolutional network）。

有关神经网络参数和超参数的高斯分布或其他形式的先验分布，见参考文献［373,398,517］相关内容。参考文献［373］中，采用拉普拉斯估计方法决定最优超参数。在参考文献［398］中，先验分布的积分和MLP中的贝叶斯学习过程采用了蒙特卡罗方法。贝叶斯学习的优点是无需校验集（validation set）即可自动确定正则因子，避免大规模神经网络中出现过拟合问题，减少预测的不确定性。参考文献［398］中指出，在隐层节点数目趋于无穷时，采用高斯先验分布的单隐层神经网络在输入—输出的函数空间上定义了一个高斯过程。因此在一些文献中不用神经网络实现，而直接使用高斯过程。[559,399,206] 虽然高斯过程为解决回归和分类问题提供了灵活的工具，但是该方法对计算能力要求极高，在现有技术下，只能将之应用于网络规模适中的问题。

5.3.2　高斯回归

在回归问题中，y的取值范围是任意的，所以在输出层中采用的最简单的激活函数是线性（实际上是恒等）函数。很自然，需要假设一个高斯概率模型，使得$\mathbf{P}(t|d, w)=\mathbf{P}(t|y(d), w)=\mathbf{P}(t|y)$是高斯的，均值向量为$y=y(d)$。进一步假设协方差矩阵是对角的，有$n$个输出节点，以下标$j$表示，从而

$$\mathbf{P}(t|d, w)=\prod_{j=1}^{n}\frac{1}{\sqrt{2\pi}\sigma_j}\exp\left(-\frac{(t_j-y_j)^2}{2\sigma_j^2}\right) \tag{5.11}$$

标准差σ_j是这个统计模型中的附加参数。若进一步假设$\sigma_j=\sigma$为常数，则当前输入的负对数似然函数变为

$$\mathcal{E}=\sum_j\left(\frac{(t_j-y_j)^2}{2\sigma^2}-\frac{1}{2}\log 2\pi-\log\sigma\right) \tag{5.12}$$

可以看出后两项与w无关，所以在优化参数w的过程中可以忽略。式中的第一项是常见的最小均方（LMS）误差，在一些时候，可不必依赖统计模型而直接用于很多实际应用。负对数似然函数$\mathcal{E}$对输出y_j的偏导数是：

$$\frac{\partial\mathcal{E}}{\partial y_j}=\frac{\partial\mathcal{E}}{\partial x_j}=-\frac{t_j-y_j}{\sigma_j}=-\frac{t_j-y_j}{\sigma}, \tag{5.13}$$

第一个等式的导出依赖于激活函数是恒等函数的假设。

总之，在带高斯噪声的回归问题中，输出激活函数应该是线性的，似然误差

函数是LMS误差函数（针对不同的j，归一化因子σ_j可能不同），$\mathcal{E}$对输出层的输入总量的偏导数具有如$-(t_j-y_j)/\sigma_j=-(t_j-y_j)/\sigma$的简单表示形式。

5.3.3 两类别分类

考虑仅具有两类别A和$\overline{A}$的分类问题。给定输入d，目标输出是0或1。最自然的概率模型是二项式模型（binomial model），则神经网络的单一输出表示输入为类别A和$\overline{A}$的概率，该概率和对应于指标函数的期望。可以使用sigmoidal激活函数进行计算。这样有：

$$y = y(d) = \mathbf{P}(d \in A) = \mathbf{P}(t|d, w) = y^t(1-y)^{(1-t)} \tag{5.14}$$

和

$$\mathcal{E} = -\log \mathbf{P}(t|d, w) = -t \log y - (1-t)\log(1-y) \tag{5.15}$$

这是输出真实分布和观测**分**布间的相对熵，并且：

$$\frac{\partial \mathcal{E}}{\partial y} = -\frac{t-y}{y(1-y)} \tag{5.16}$$

特别地，如果输出激活函数是logistic函数，则

$$\frac{\partial \mathcal{E}}{\partial x} = -(t-y) \tag{5.17}$$

因而，在两类别分类问题中，输出激活函数是logistic函数；似然误差函数是预测值分布和目标值分布间的相对熵。对应每一样本，$\mathcal{E}$对输出层的输入总量的偏导数具有$-(t-y)$的简单表示形式。

5.3.4 多类别分类

更一般地，考虑带有n个可能类别$A_1,\cdots, A_n$的分类问题。给定输入d，目标输出是有一个元素为1，其余元素都为0的n维向量形式。最简单的概率模型是多项式模型（multinomial model）。对应的神经网络具有n个输出节点，每个节点的输出给出了输入向量对应于该类别的概率。这样：

$$\mathbf{P}(t|d, w) = \prod_{j=1}^{n} y_j^{t_j} \tag{5.18}$$

通常地，其中$t_j=t_j(d)$，$y_j=y_j(d)$。对于每一个样本有：

$$\mathcal{E} = -\log \mathbf{P}(t|d,w) = -\sum_{j=1}^{n} t_j \log y_j \tag{5.19}$$

这是真实分布和观测分布间的相对熵，并且

$$\frac{\partial \mathcal{E}}{\partial y_j} = -\frac{t_j}{y_j} \tag{5.20}$$

特别地，如果输出层节点是归一化指数形式，则对应每一输入d_i，有

$$\frac{\partial \mathcal{E}}{\partial x_j} = -(t_j - y_j) \tag{5.21}$$

其中x_j是到第j个归一化指数函数的总输入量。

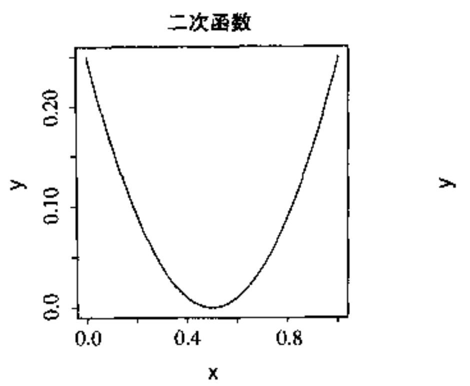
由此看来，在多类别分类问题中，输出层的激活函数应该是归一化指数函数。似然误差函数是被预测值分布和目标值分布间的相对熵。对应每一样本和每一类别，$\mathcal{E}$对到输出层的输入总量的偏导数具有$-(t_j-y_j)$的简单表示形式。

5.3.5　一般化的指数族类型

事实上，当似然函数属于指数族分布类型时，得到的结果与前面所得到的结果类似（见附录A和参考文献［384,94］）。指数族分布包含很多种最常见的分布类型，如高斯、伽玛、二项、多项、指数、贝塔、泊松、负二项分布等。针对每一种分布，可以选择合适的输出激活函数$y=f(x)$，使得$\mathcal{E}$对第j个输出节点的输入总量的偏导数$\partial \mathcal{E}/\partial x_j$具有简单表示形式。对应每一个样本，该偏导数正比于$(t_j-y_j)$，即目标输出$t_j$与实际输出$y_j$间的差别。

可以看到，统计理论允许为输出层构造合适的激活函数，并构造合适的误差函数来衡量网络的性能。然而隐层的设计依赖于具体问题，没有一般的模式可循。因为上述的理论体系在近几年中才逐渐形成，所以并非所有的神经网络设计者都严格遵循这一理论进行神经网络的设计开发（包括以下章节所提及的许多例子）。很多研究者甚至在两类别分类问题中依然使用LMS误差函数，虽然理论上相对熵误差函数更为合适。

“如果所使用的理论不合适，如何得出合理的结果呢？”上述的简单例子给出了这一问题的答案。为了更好地说明问题，假设在两类别分类问题中，希望学习的概率为$p=0.5$。对于在［0, 1］中取值的每个x，LMS误差为$(0.5-x)^2$，所以相对熵为$-0.5\log x-0.5\log(1-x)$。图5-3中绘出了这两个函数的曲线。这两条曲



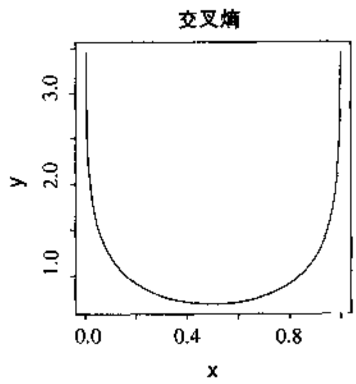


图5-3 一维二次函数和交叉熵误差函数的比较

函数对应的目标值为0.5。注意两条曲线值域的不同：交叉熵在x=0和x=1处趋于无穷大。

线都是下凸的，在p=0.5处达到最小值。主要的不同点在于动态区域范围：与相对熵不同，LMS误差是有界的。当许多样本误差叠加的时候或在样本学习过程中，动态区域的差别将会显得较为重要。

5.4 反向传播学习算法

在我们将要讨论的大多数实际应用中，神经网络参数的MAP或ML估计是采用梯度下降学习算法实现的（见参考文献［26］）。梯度的计算可以顺次依照神经网络的拓扑结构完成。由输出层回馈至输入层，误差信号依照神经网络连接反向传播，依次更新权重。更精确地，在神经网络参数的各种在线训练过程中，参数w_{ij}满足

$$\frac{\partial E}{\partial w_{ij}}=\frac{\partial E}{\partial y_i}\frac{\partial y_i}{\partial w_{ij}}=\frac{\partial E}{\partial y_i}f_i'(x_i)y_j \tag{5.22}$$

因此，梯度下降学习方程为三项之积，即

$$\Delta w_{ij}=-\eta\frac{\partial E}{\partial w_{ij}}=-\eta\varepsilon_i y_j \tag{5.23}$$

其中η为学习率，y_j为神经元的输出，$\varepsilon_i=(\partial E/\partial y_i)f_i'(x_i)$称做反向传播误差。反

向传播误差可由下式递归计算得到：

$$\frac{\partial E}{\partial y_i}=\sum_{k\in N^+(i)}\frac{\partial E}{\partial y_k}f_k'(x_k)w_{ki} \quad (5.24)$$

反向传播算法的特点是误差从子节点向父节点传播。反向传播算法和EM、模拟退火算法一样，是在对MLP结构进行MAP估计中被广泛应用的算法。这些算法应用在网络结构学习方面比较理想，但随着问题维数的增大，它们越来越低效。

现在可以回顾神经网络在分子生物学方面的重要应用了。相关主题的学习见参考文献［432,571,572］。

第6章　神经网络：应用

神经网络很早就在生物序列分析领域中获得了应用。1982年，以氨基酸序列作为输入向量的感知器已用于核糖体结合位点的预测。[506]斯托墨（Stormo）和他的同事们发现，在寻找大肠杆菌转录起始位点方面，感知器算法优于以往那些基于规则推断的算法。[507]不含隐节点的感知器具有推广能力，可以在序列中找到训练集（training set）中未出现的转录起始位点。

对很多序列识别问题而言，线性的网络结构并不十分有效。直到多层感知器的反向传播训练算法于1986年开始广泛应用[456]，尤其是1988年钱和塞诺斯基发表关于蛋白质二级结构预测的论文[437]后，神经网络才得到足够的重视和真正广泛的应用。这篇以及随后的其他几篇论文[78,262]中使用的神经网络，都是以NetTalk多层感知器结构为基础的，[480]NetTalk可以由输入给神经网络的英文文章中的字母来预测相关音素，以满足语音合成和文章机器阅读的需要。只需要把输入的字母改成相应的氨基酸或核苷酸字符，就可以立即将这种方法应用于序列分析领域。同样，音素的编码可以很容易地转换成结构类别，例如：蛋白质的二级结构类别（螺旋、折叠或卷曲）或不同的功能类别（如结合位点、剪切位点或转录后修饰的残基等）。

在本章中，将首先对应用于蛋白质和核苷酸分析的一些早期工作做个综述。然后详细阐述当前研究中的一些例子，这些例子所用的方法在训练法则或在神经网络结构上具有优势，尤其是能够将不同的神经网络结合在一起而产生出更好的预测效果。本章的目的不在于囊括神经网络的整个应用范围，其他的最新进展的综述见参考文献［432,61,77,320,571,572］。

6.1 序列编码和输出表示

在将神经网络应用于分子生物学之前，必须首先讨论一下序列输入编码这一重要的问题。无论采用什么样的预测方法，输入量的编码表示形式十分重要。如果编码表示形式选择得当，能够揭示特定问题的本质，那么问题或多或少可以得到解决，或至少可采用简单的线性方法得以解决。在MLP中，最后一个隐层到输出层的输出所传递的输入信息应该以线性可分的形式存在。很明显，如果输入的编码表示形式没有增加非线性程度，则问题容易处理得多。

有人也许认为使用一套与所研究问题可能相关的物理化学特性来对序列进行“实际的”编码，要比使用由信息理论规则提炼的抽象编码形式更为有效。[137]然而同大多数预测方法的信息约简特性一致（见1.4节），由于神经网络在输入特征到达输出层前会滤掉大部分额外信息，所以一些额外信息的引入并不一定会提高算法的性能。

在MLP的训练过程中，神经网络使用超平面（hyperplane）将输入空间划分为不同的决策区域。由于序列是以数字量化的形式编码的，所以在数字量表示的输入序列所定义的空间上，输入序列的编码表示形式对隐层所决定的超平面的位置有很大的影响。

在许多序列分析中，输入常取长度为W的窗，这个窗覆盖了一个或几个相关序列片断。一般来说，窗的位置是对称的，以保证上游和下游序列具有相同的长度。但在个别案例中，使用不对称窗要比使用对称窗的效果好。在识别信号肽剪切位点（第6.4节）和mRNA前体中内含子剪接位点（第6.5.2节）时，使用不对称窗就比使用对称窗的效果好得多。这两种类型的序列（N端的蛋白质分选信号和非编码的内含子DNA序列）最终是要被切去的，因此尽量保留序列内部关于蛋白质合成过程的特征信息是有意义的，这样可以使最后得到的成熟蛋白所受约束最少。一些窗口带有空隙，其中的序列并不紧密相连，已经专门用于识别启动子、DNA转录起始位点、蛋白质β折叠伴侣。[268,46]这类窗口还可以基于残基的上下游序列，预测两个氨基酸的距离约束。[368,174]

窗W的每个位置上，有$|\mathbf{A}|$种不同的可能单体出现。最常用的编码表示形式称为正交编码（orthogonal）（为区别于分布式编码，也可称为局域编码），此时字符$\mathbf{X}_1, \mathbf{X}_2, \cdots$用正交向量（$1, 0, \cdots, 0$），（$0, 1, \cdots, 0$），……编码。这种编码表示形式的优点在于不引入任何单体间的代数相关。不完整的氨基酸序列窗中所出现的N、C端位置常使用专门的特征字符进行编码。有时，这个特征字符也可用于序列中未知类型单体的编码，当然未知类型单体也可以采用全为0的字符串的形式编码，

这样它们对网络的输入层无任何影响。

这种稀疏编码方式明显有浪费资源的缺点，因为它需要输入层的规模为|A|×W。|A|个字符理论上最少只需要用$\log_2|A|$个二值节点编码。进一步说，如果MLP的输入层取值连续，则一个节点就能编码所有可能的字符。在预测问题中，这类压缩编码方式势必会大大增加非线性的因素。如果所有的氨基酸编码取值都在［0, 1］区间中，无论以什么样的顺序将序列元素映射到该段区间内，这样产生的许多序列元素间的数学相关性并无任何生物学相关意义。

显然，不同的编码方案必然会对输入窗所处空间的复杂度、神经网络的结构和学习的难易程度产生影响，建模者通常要在这些方案中进行折中。在目前该领域大部分最好的工作中，正交编码成为效果最佳的编码方式。面对一个较为复杂的编码序列输入，无论是否采用了正交编码，神经网络会将输入向量映射为维数等于隐层节点数的空间中的点，然后进一步将该点映射到具有更少节点，通常仅仅一个节点的输出空间，从而过滤掉其他冗余信息。如果相对于残基的物理化学特性，输入层的输入向量包含了太多的额外信息，而这些信息与目标输出的关联又不紧密，则势必会增加神经网络识别分类的难度。这种情况下，最好增加隐层的数量，以保证能够剔除这些额外信息，并从中提取相应的关联特征。由于缺少其他更好的解决办法，所以在这种情况下，使用正交编码效果会更好。

如果选择采用实数量化的残基疏水性、电荷数、体积等指标进行编码，必须注意其可能对所定义的输入空间产生的不利影响。将原始序列片断经过预处理后进行编码，要比直接将输入残基编码的效果要好得多。对原始序列进行预处理时，一般需要对窗口中出现的特定词做词频统计，计算整个窗口或分别计算不对称窗中左右半个窗口的平均疏水性等指标。另一种可能的有趣方式是让神经网络学习构造自己的编码形式，这将在下面的一个例子中使用。另外一个例子说明了在蛋白质二级结构预测中，二元词组编码（binary word encoding）方式提高了预测的性能。[313,548,17]在这个例子中，可以进一步利用由模拟退火算法产生的优化编码，来推测和挖掘与二级结构形成相关的物理化学特性。

降低预测的非线性程度的一种重要的策略，就是将基于单体的编码形式转变成基于二元组或三元组的编码形式。对于核苷酸而言，二元组和三元组对应于16和64个字符表示。在大量的生物识别问题中，存在大量二元或三元关联，因此使用二元组或三元组的编码方式所带来的效益远远大于相应输入空间维数增大所带来的负效果。在DNA中，碱基对堆积（base pair stacking）对螺旋稳定性的热力学贡献最大（超过了碱基互补的贡献）。例如，在RNA-RNA相互作用识别中，碱基组关联在相邻碱基对的堆积能有其物理意义。[112]蛋白质的二肽分布与结构空

间阻隔、转录动力学和其他纯生化指标也有很强的联系。

如果DNA和RNA序列以二元组或三元组的形式编码，多聚体重叠（multimers overlap）的情况就有可能发生。多聚体的稀疏编码方式保证了序列数据不再包含任何先验关系。以重叠三元组编码序列的优点是隐层可以直接获得每个核苷酸的上下游序列信息，而不必再依靠训练过程推断。

另一种减小（或者在一些实例中增加）预测非线性程度的策略是将一些单体简并成新的字符形式，从而获得一套新的字符集，以便使所寻求的特定模式和背景信息形成更鲜明的对照。[306]简并的字符集可以以正交向量形式编码，从而降低了输入空间的维数和神经网络中可调整参数的个数。可以根据物理化学特性或在蛋白质家族的进化研究中所建立的突变速率，做出有意义的字符简并。表6-1中列出了以前用到的一些简并编码形式，这些编码形式或者基于各种单体的初始字符表达，或者基于从实验数据中观察到的单体结构或功能信息。

表6-1　生物分子单体的简并编码

分　子	分组类别数	分组类别
DNA	2	嘌呤/嘧啶：R=A，G；Y=C，T
DNA	2	氢键强/弱：S=C，G；W=A，T
DNA	2	按生化特征： 酮类的，K=T，G/氨基的，M=A，C
蛋白质	3	按结构特性： 不确定的（Ala，Cys，Gly，Pro，Ser，Thr，Trp，Tyr） 外部的（Arg，Asn，Asp，Gln，Glu，His，Lys） 内部的（Ile，Leu，Met，Phe，Val）
蛋白质	8	按化学特性： 酸性的（Asp，Glu） 脂肪族的（Ala，Gly，Ile，Leu，Val） 氨基化合物（Asn，Gln） 芳香族的（Phe，Trp，Tyr） 碱性的（Arg，His，Lys） 羟基（Ser，Thr） 亚胺的（Pro） 硫化的（Cys，Met）
蛋白质	4	按功能特性： 酸性和碱性（类按化学特性的类别划分） 疏水非极性（Ala，Ile，Leu，Met，Phe，Pro，Trp，Val） 极性不带电荷（Asn，Cys，Gln，Gly，Ser，Thr，Tyr）

（续表）

分　子	分组类别数	分组类别
蛋白质	3	按电荷性质： 酸性和碱性（类按化学特性的类别划分） 不带电的（所有其他氨基酸）
蛋白质	2	按疏水特性： 疏水的（Ala，Ile，Leu，Met，Phe，Pro，Trp，Val） 亲水的（Arg，Asn，Asp，Cys，Gln，Glu，Gly，His，Lys，Ser，Thr，Tyr）

这些简并编码有的基于单体的初始描述，有的取自基于结构和功能信息的单体的统计特性描述。可以将氨基酸随机划分为使序列间相似性最大的k类。更深入的内容见参考文献［306］。

最近的研究表明，蛋白质可以最大程度地保持其折叠结构，即使组成蛋白质的氨基酸数目由传统的20种减少到5种。[443]除了结合位点附近的一些位置，有15种不同类型的氨基酸可以由某些残基替换，这些残基来自由数量更少、更有代表性的5种氨基酸组成的编码组（I、K、E、A和G）。如果按这一思路进一步将氨基酸减少到3个，效果则不理想。这意味着在早期的进化中，蛋白质能够依赖一组数量少得多的氨基酸单体集合获得稳定的折叠结构。需要指出，这类作为代表的简并字符并非是普适的：许多缺少了半胱氨酸的蛋白质根本无法发挥正常的功能。在生物信息学方法中，使用较小的字符集（表6-1）重新编码序列看上去好像只是单纯的计算技巧，但是实际上，在编码工作中也可利用更多的关于“基本”氨基酸的实验性工作，来构建更简化的序列空间集合以适应有限的数据。这里所提及的简并策略的灵感源自蛋白质进化变异的研究。这一策略为如本章下一节所描述的蛋白质结构预测方法提供了一类附加信息，从而改善了方法的预测性能。

在其他应用中，编码过程不必保持原有序列中残基的连续顺序，而是对整个序列或序列的一个大片断进行预处理，得到的编码向量可作为神经网络的输入信息，例如使用400个氨基酸二元组的频率信息来预测蛋白质的折叠类和家族关系时，就可使用以上预处理方法。[179,18]在内含子、外显子分类中，也可使用间接的编码方式：将六元组统计信息、GC成分信息、序列词表信息以及其他一些指标整合为输入向量，作为神经网络的输入信息。[529]

在下面所描述的神经网络的应用中将提到，一种好的输出表示和后期处理策略也是十分重要的。在大多数的实际应用中，智能化的后期处理可能比选择最优网络结构更重要，这个最优网络结构优化准则是使输出神经元的输出量量化的推广误差最小。一般地，输出神经元的个数直接对应于输出类别的数量，也使用正

交0、1向量的稀疏编码形式表示。输出表示形式和后期处理形式的设计依赖于问题的生物学背景特征。例如，若知道蛋白质中的α折叠所需要的氨基酸长度最小为4这一个先验知识，则在预测结果中长度小于4的“折叠”就会被剔除，从而提高了整体预测性能。又如，若知道一段序列中只包含单一的给定类型的功能位点，比如N端信号肽的剪切位点，就可以更恰当地设计阈值界定规则，以保证真实位点的高识别率和降低样本的假阳性率。关于连续网络误差和离散分类误差之间关系的讨论见参考文献［90］。

6.2 序列相关性与神经网络

即使考虑了氨基酸的相似性，许多序列的结构功能单元在序列位置上也不是保守的。众所周知，即使采用氨基酸残基对应位置的比较评估和量化所得到的序列相似性很低，蛋白质的结构也可能高度保守。蛋白质结构的形成，无论是局部结构还是全局结构，都不仅仅依靠各个位置上的残基的独立贡献，而是依靠残基序列间的协同作用完成的。

不但对全长的蛋白质如此，局部区域的蛋白质也是如此，例如由特定激酶识别的磷酸化位点motif（超二级结构模体或一段具有特色功能的生物序列，前者为主）。即使那些与相同的激酶发生作用的线性motif，其序列的模式也有很大不同。[331] 考察这些序列片断的局部结构［借助蛋白质数据库（PDB）中结构已知的蛋白质］，可以发现它们即使在氨基酸组成上是多样的，在结构上也可能是高度保守的。[74]

由于具有将不同的输入值相关联的能力，神经网络技术具有检测序列间协同性的潜力。实际上，人们猜测：在训练过程中得到的权重间的协同性，恰恰反映了输入单体间的关联性，而这种关联性又与神经网络所执行的预测任务密切相关。

神经网络这种建立序列不同位置关联性的能力，与人脑依赖上下文推断语句中不同字符的能力相类似。例如，在包含有四个“a”的句子“Mary had a little lamb”中“a”发三种不同的音，联想关联能力在这类发音问题中体现得很显著。[480] 另一个说明这种关联能力的实例见图6-1所示。在图中，只要接收映射到视网膜上的信息的大脑接受过阅读英文、理解英文文本顺序结构的训练，同一个符号就可以理解成不同的意思。

正是这种能力才使得神经网络在序列分析中获得了极大的成功，因为它在一定程度上弥补了权重矩阵和HMM的一些不足。由于在输入层编码的序列信息可以来自给定序列的不同位置，所以神经网络技术并不仅仅局限于对局部相关性的

HUMAN

图6-1 人类阅读的联想关联能力

阅读英文的人通常能对这个单词中出现的那两个形状相同的字符做出不同的识别：前一个是*h*，而后一个是*a*。在生物序列分析中，当结构和功能特征更多地来源于序列中的协同联系而不是由单独的核苷酸或氨基酸决定时，这种类似的信息处理能力就显得十分必要了。神经网络技术具有检测短程、长程序列相关的能力，从而弥补了传统的HMM序列分析的不足。

分析。[368] 但正如以下章节所讲述的，大多数应用仍集中于局部和线性序列片断的分析。

6.3 蛋白质二级结构预测

当人们在计算机屏幕上观察蛋白质骨架结构的图形时，可以马上看出一种重复结构形式的局部折叠规则性。有两种二级结构（通过骨架中的氢键保持其结构），在用X光衍射法得到其一级结构之前，就已经有理论预测出它们的存在了。对二级结构的类型尚没有规范的定义，但表示每个氨基酸残基拥有成对二面角的Ramachandran图显示，特定的二面角区域基本上代表了实际蛋白质的特定折叠区域。一个区域与螺旋相对应，此时骨架中的氢键连接第i个和第i+4个残基；另一区域与β折叠对应，其中氢键以平行或反平行的方式连接两段序列片断。

这类结构中存在的序列偏好和序列间关联，使得二级结构预测问题成为计算分子生物学的经典问题之一。[362,128,129,196] 无论早期的研究[437,78,262,370,323] 还是现今的先进方法[453,445]，都使用了不同类型的神经网络结构来研究这一问题。

在实验手段所测定的三维结构中匹配二级结构的种类，是非常繁锁的。现在，这个工作的大部分内容由广泛使用的DSSP程序完成。[297] DSSP程序通过骨架原子的三维坐标来分析潜在的氢键重复结构模式。另一个可以完成该匹配功能的程序是STRIDE程序，其同时使用了氢键能量和骨架的二面角信息，而不仅仅是氢键信息。[192] 而DEFINE程序的处理思路是，使用不同的距离矩阵来计算蛋白质的原子间距离与理想的二级结构的原子间距离之间的差异。[442]

以上这些程序都不很完善。在坐标数据精度有限的情况下区分螺旋和折叠，

其算法并不是很简单。另一个难点在于，量子化学不能给出非常准确的氢键强度解析表达式。更为理想的处理方法，不应仅仅把注意力集中在问题的可视化和拓扑结构构建上，还应试图寻找一种预测能力更好的二级结构匹配方案。一种压缩的预测方案剔除了一些螺旋和折叠结构，从而获得了近乎完美的预测效果。这种方案十分有价值，尤其是在三级结构预测中，通常使用二级结构的预测结果作为预测的起点。

6.3.1 利用多层感知器（MLP）预测二级结构

钱和塞诺斯基的早期工作中使用的基本网络结构是带有一个隐层的全连接多层感知器。[437] 输入窗口长为W（奇数），最优长度一般为13个氨基酸。输入采用了正交编码的方式，字符集大小为$|\mathbf{A}|=21$，对应于20个氨基酸和1个终止符号（用于编码N端或C端的不完全窗）。这样，输入层有$13\times21=273$个节点。通常采用的隐层包含40个sigmoidal类型的节点。因此，这一网络结构的总参数个数为$273\times40+40\times3+40+3=11\ 083$个。输出层有3个sigmoidal类型的节点，其正交编码分别针对α螺旋、β折叠和卷曲三种类别。不同的输出表示在窗口中间位置上对应残基的不同归类（共分为三类）。类别的归属是由输出最大的输出节点按照“胜者通吃”原则（the winner-take-all principle）决定。“胜者通吃”原则的采用，为输入和最终输出分类之间的关系增加了额外的非线性特征。因而采用这一原则的不带隐层的神经网络，不再是完全线性的了。序列输入在内部隐层中的向量表示形式需要被线性平面分割的要求不高时，也会引入这种原则。只要正确的输出节点到分类超平面的距离最小，就在一定程度上允许带有一定扰动的输入样本向量点落在其他类别的区域中。

网络初始化时，权重初值为在区间［-0.3, 0.3］上均匀分布的随机变量，然后使用基于LMS误差函数的反向传播算法训练网络。（注意：采用相对熵误差函数的归一化指数类型的输出层更为合适些。）训练集的大小约为20 000个残基，它们取自Brookhaven蛋白质数据库（PDB）。因此参数数目与样本量的比值相当高，大于0.5。现今，越来越多的蛋白质结构通过实验测得，相对应的二级结构匹配数据库的规模也越来越大。

利用蛋白质序列进行训练时，训练集的输入窗以随机顺序输入神经网络，避免了使用连续输入窗给性能带来的影响。使用这种神经网络结构，模型的预测正确率从随机分类时的33%上升到60%，然后才会出现过拟合。更精确地讲，整体正确率为$Q_3=62.7\%$，对应相关系数为$C_\alpha=0.35$，$C_\beta=0.29$，$C_c=0.38$。[382] 由于自然界的蛋白质中出现的螺旋、折叠、卷曲结构的数量不等（大约为0.3：0.2：0.5），

所以仅仅通过窗口的正确预测率尚不能充分说明预测算法的性能。于是，引入一个新的性能度量——相关系数，[382]其综合考虑了正、负样本预测正确、错误数目间的影响与联系：

$$C_X = \frac{(P_X N_X) - (N_X^f P_X^f)}{\sqrt{(N_X + N_X^f)(N_X + P_X^f)(P_X + N_X^f)(P_X + P_X^f)}} \tag{6.1}$$

其中X可以表示螺旋、折叠和卷曲类别之一或由其中的多个融合为一的类别。P_X和N_X为预测正确的正样本和负样本数，类似地，P_X^f 和 N_X^f 为预测错误的正样本和负样本数。完全正确的预测对应$C(X)=1$，完全错误的预测对应$C(X)=-1$（更详尽的讨论可参见6.7节）。

研究人员进行了一系列测试网络结构及其他特征变量的实验，他们得出如下结论：将输入增加为大于13个氨基酸的长度或增添诸如氨基酸疏水性之类的其他额外信息，并不能提高算法的预测性能。同样，使用更好的二级结构分类方法、更高阶甚至反馈的神经网络或剪枝法（pruning method）等，都不能提高预测的性能。

通过在先前的网络结构上串联另外一个神经网络，它利用前一层结构的三个节点的某些相似的输出值以及相邻节点输出值的相关性，可以使预测性能提高。后一个神经网络的输入窗长度为13，对应前一个网络的13个连续输出。这样，后一个神经网络的输入层节点为13×3个，另外还具有一个包含40个节点的隐层和通常为3个节点的输出层。使用这种串联结构，整体预测性能达到Q_3=64.3%，对应相关系数为C_α=0.41，C_β=0.31，C_c=0.41。训练结束时，后一个网络去除了前一个网络输出中不连续的预测结果，使预测结果更为合理。从以上的分析结果可知，只使用“局部”算法（这里的“局部”针对预测算法的输入窗长度而言），最终可达到的预测性能的上限为略大于70%。在1988年，这些结果看起来较包括Chou-Fasman算法[129]在内的其他算法好得多。后来随着数据量的不断增长，各种更为先进的神经网络算法的性能得到了很大的提高，但Chou-Fasman算法的性能改善却不很明显。[549]从上面的分析可以看出，现在的一些二级结构预测算法已经超越70%的预测水平，有些算法的预测水平甚至接近于80%。

6.3.2 基于进化信息和氨基酸构成的预测方法

大多数后继的使用神经网络的二级结构预测工作[78,262,323,505,451,452,290,427]都以上述的网络结构为基础，有时会与诸如Chou-Fasman规则[129]之类方法相结合。[582,377]

一个很有意思的实例是使用Chou-Fasman规则来初始化神经网络。[377]这一基于知识的神经网络与将规则直接编码到权重中的神经网络在性能上类似。因此可以使用PDB中的实验数据来训练所添加的额外的自由连接。所有规则中都不包含的输入序列和结构分类间的关系，将在训练校正额外参数的过程中得到处理。虽然其性能只比钱-塞诺斯基网络稍有提高，但这一结构仍有引人之处，因为它很容易考察网络的权重。与Chou-Fasman规则相比，正如期望的那样，其性能大大提高。

参考文献［505］中比较了MLP和贝叶斯这两种方法。在这项研究中，贝叶斯方法是以人为假设蛋白质每个位置上的氨基酸出现概率与周围位置上的氨基酸类型独立为基础的。尽管如此，其预测正确率仅略低于上述构造的神经网络。研究人员构造了另外一种神经网络，其中输出神经元直接表示结构类别的条件概率。而概率形式的方法允许引入新的目标函数——互信息（mutual information），互信息将作为预测正确率度量的相关性概念转化为一种有用的训练方法。虽然概率形式的方法在正确率上与其他方法相差无几（同样利用均方误差函数），但它的训练集的正确率明显比其他方法高。即使在可调参数数量不变的条件下，与使用均方误差指标相比，使用互信息指标以牺牲卷曲结构的预测正确率为代价，提高了对螺旋和折叠结构的预测正确率。

虽然基于不同数据集的算法测试很难比较，但罗斯特和桑德设计的PHD预测服务器[451,452,453]所得到的研究结果较以往的各类方法，在预测性能上获得了最显著的提高。在1996年举行的Asilomar CASP2（蛋白质结构预测技术评判）竞赛中，该方法在二级结构预测方面明显优于其他方法。[161]研究人员进行这一专门的试验是为了尝试评判蛋白质二级结构黑箱预测技术的现有发展水平。在预测者眼中，一些正在解析结构的蛋白质序列是十分合适的预测处理对象。对于预测竞赛第一阶段所提供的未知测试集，PHD方法的预测正确率达到74%。竞赛第一阶段包含关联、二级结构和分子模拟预测，而这一阶段的竞赛是公认最难的，因为对竞赛者而言，所有的先验知识仅仅为氨基酸的一级序列结构。

基于单一序列的三类别二级结构预测的正确率低于65%~68%。20世纪80年代中期，三类别的预测正确率达到了50%~55%，更高级的神经网络算法和增长的数据集又使正确率提高到65%，人们曾经认为这个水平无法逾越。与现今的其他较成功的方法类似，PHD的核心思想是认识到序列家族比单一序列包含更多的有用信息，因此预测算法不应只提取13~21个连续残基的序列片断中所包含的局部信息。这一思想在先前的多重序列比对的研究中有所体现，具体可见参考文献［587,139,60］。

使用进化信息使预测正确率进一步提高至大于72%，相应的相关系数为C_α=0.64，

C_β=0.53。使用进化信息进行预测有四个步骤：第一步，扫描已知序列数据库，使用比对方法寻找类似序列簇；第二步，利用与序列长度有关的阈值对序列进行筛选，寻找有显著相似性的序列个体；第三步，基于所有可能的三维同源性，完成一系列氨基酸间的替换；第四步，将完成氨基酸替换后得到的新序列谱用于预测。

PHD方法是第一种使用250条独特的蛋白质链进行交叉验证并证明其二级结构三类别预测正确率大于72%的方法。[451,452,453] 正如图6-2所示，该方法是将替换后的新序列谱和源自多重序列比对的附加信息以及蛋白质的氨基酸组成整合在一起，作为输入向量赋予神经网络。输入向量不再基于传统的单一序列的正交编码，而是基于对与待预测序列高度同源的序列簇进行多重序列比对所获得的序列谱（氨基酸在不同位置出现的频率向量）的。在图6-2所示的例子中，序列谱共包含5个序列。序列中小写字母表示比对序列的间隙。蛋白质的每一特定位置对应有20个值（1列）以及3个附加值：间隙个数、插入个数和保守权重。将13个相邻列联合作为输入。"L"（环）类别等同于其他绝大多数研究文献中出现的卷曲分类类别。整个二级结构预测系统包含3层：2个神经网络层和1个对独立训练的神经网络结果取平均的计算层。

这一研究工作中的序列谱集合取自HSSP数据库。[471] HSSP是一个融合了结构和序列信息的二级数据库。对于每个源自PDB的已知三维结构的蛋白质，该数据库给出了所有可能的同源性多重序列比对和该蛋白质家族的序列特征。

神经网络的反向传播训练过程可能是平衡的或非平衡的。在大型的、低同源性的蛋白质数据库中，螺旋、折叠和卷曲结构的数量比例是30%、20%和50%。在非平衡的训练中，13个氨基酸长度的向量以相同的概率随机出现。在平衡的训练中，不同类别的训练样本个数常常是相同的。这意味着螺旋和折叠类的样本为卷曲类样本的2倍。在最终的预测系统中，人们使用兼具这两种训练方案的网络。由平衡方案训练的神经网络对折叠类的预测更可靠。

网络的许多其他细节对于提高整体预测正确率，尤其对于提高折叠类的预测正确率和提高二级结构片断而非单个残基的预测正确率，是十分重要的。对40%的高可靠性残基进行预测，该方法正确预测率约达90%，这个正确率与同源模建方法持平。整体预测正确率中将近10个百分点的提高要归功于进化信息的引入。

显而易见，钱-塞诺斯基网络结构的一个重要缺陷是存在过拟合的问题。罗斯特和桑德使用了相同的基本结构，但应用了两种策略来解决过拟合的问题：第一种策略是提前结束；第二种策略是将用不同输入和不同学习算法独立训练的不同网络做整体平均。[237,340] 但他们的工作中最具创新性的方面在于多重序列比对的运用，即用序列谱（亦即源自多重序列比对的位置相关的频率向量）取代原始

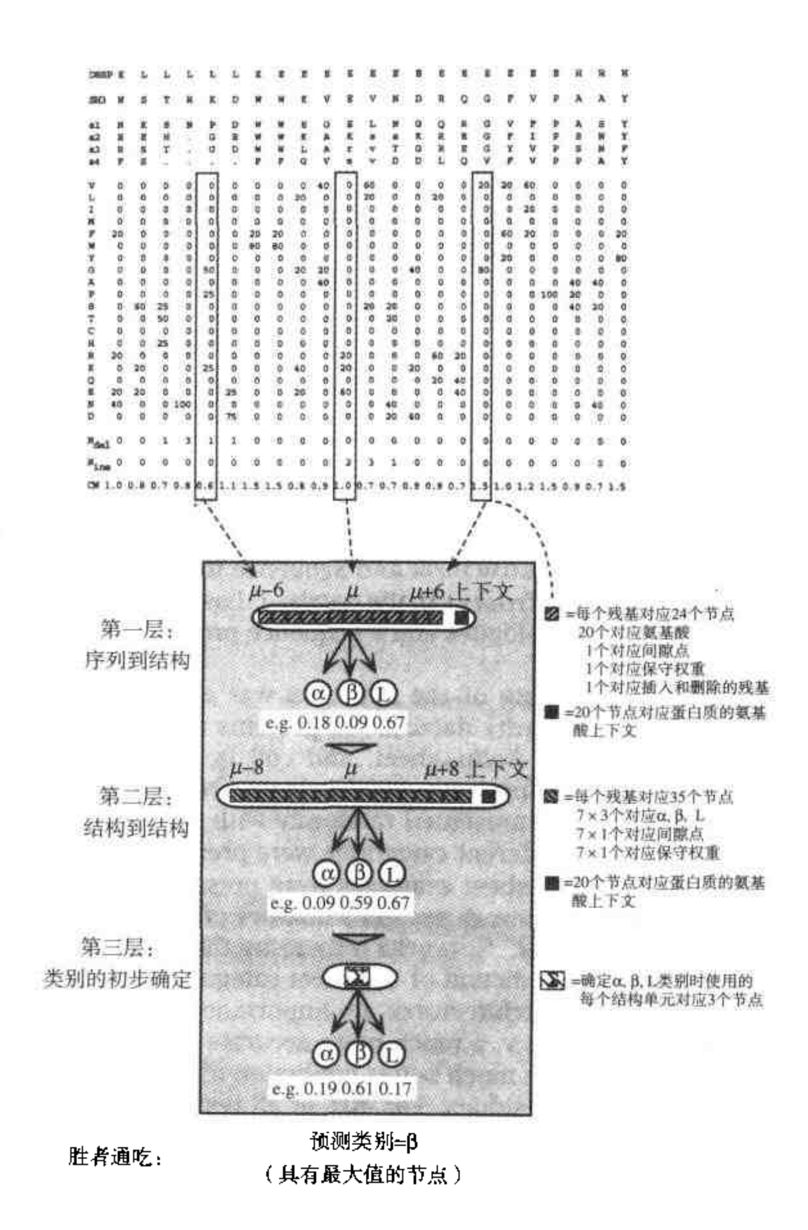


图6-2 罗斯特和桑德提出的二级结构预测方法PHD的网络结构图

输入向量不再是传统的序列的正交编码，而是来自对与待预测序列高度同源的序列簇进行多重序列比对所获得的序列谱（每一列分别代表氨基酸在相应位置出现的频率）。

的氨基酸序列向量作为网络的输入。之所以如此，是由于多重序列比对比单一序列包含了更多的二级结构信息，而二级结构具有比一级结构更强的保守性。

6.3.3 网络模型集和自适应编码

里斯（Riis）和克罗（Krogh）研究了二级结构预测的另一种独特的神经网络方法，[338,445] 他们通过精心设计神经网络结构避免了过拟合问题。该方法由四个主要部分构成：第一部分，在早先的网络结构中引入大量参数是由于输入层的维数比较大。对氨基酸进行自适应编码，即通过神经网络寻找一种输入字符的优化、压缩编码表示形式，可大大减小输入层的维数。这可以通过使用局部或分布式编码加以实现，即每个氨基酸编码利用M个取连续值的节点进行编码。更详细地说，先使用20个节点进行正交编码，全零向量表示N端或C端的空白符号。此时输入层大小为$W \times 20$，以特定的连接模式与带有$M \times W$个隐层节点的第一隐层连接。输入层序列的每个位置与M个sigmoidal节点连接，这种连接必须是恒定的，即在序列不同的位置，其连接值是恒定不变的。在神经网络术语中，这一技术称为“权重共享”。在图像处理问题中，这组固定连接被定义为卷积滤波核（kernel of a convolution filter）。在对取相同值的权重进行调整时，只需将权重修正值相加求和，即可很容易地将反向传播算法应用于“权重共享”网络的参数学习中。这样字符集中的每个字符就编码成M个取连续值的节点。在模式识别中，这样的M个节点可看做一个特征提取器。值得注意的是，利于问题解决的特征是在学习过程中优化和提取得到的，而不是事先已知的。无论窗长W是多少，输入层与该表示层之间的自由连接的数量（包括偏倚在内）仅为$21 \times M$个。与前一网络结构的第一层中出现的10 000多个参数相比，参数的数量大大减少。在里斯和克罗的研究工作中，取M=3，W=15。

第二部分，里斯和克罗为三类别中的每一类分别设计了不同的网络。在α螺旋的识别中，通过在第一隐层和第二隐层之间建立三残基周期性连接，将螺旋的周期特性加以利用（见图6-3）；第二隐层与输出层建立全连接关系。在β折叠和卷曲的识别中，第一隐层与一般为5~10个节点大小的第二隐层建立全连接关系；第二隐层与输出层建立全连接关系。这样，一个α螺旋网络一般共有160个可调参数，β折叠或卷曲网络一般包含300~500个可调参数。在分别训练这些网络时，使用了平衡学习的训练集，使正、负样本数相同。

第三部分，里斯和克罗使用网络模型集和过滤技术提高预测的正确率。对每个位置的每种不同类型的结构使用5个不同的网络模型。模型集中的每个网络是各不相同的，例如隐层节点数目互不相同。这样组合成的网络采用一个长度为15个连续预测点的窗口。因此，该网络输入层节点数目为$15 \times 3 \times 5$=225（见图6-4）。为了使总参数数目处于合理范围之内，需要限定每个位置每个类别（α、β和卷曲）仅可包含1个隐层节点。这样，输入被局部连接到一个包含3×15=45个节点的隐层。最后，隐层与3个归一化指数类型的输出节点全连接，来计算窗中央点的残

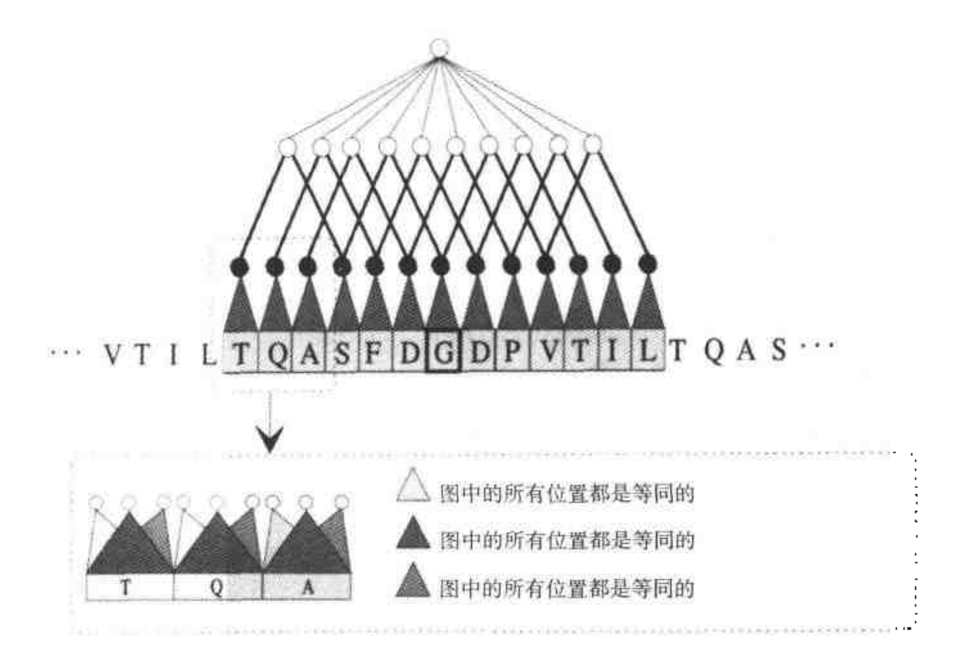


图6-3 预测螺旋结构的里斯和克罗的网络

该网络使用了局部编码策略，并引入了三残基周期。黑色圆圈表示3个隐层节点，粗线表示3个网络权重。在图的下部，带阴影的三角形表示20个共享权重，带阴影的矩阵表示20个输入节点。网络选用的窗口尺寸为13个残基，并有1个输出神经元。

基归属各类的概率。与上面的理论分析相一致，误差指标采用了负对数似然度，在此例中即为真实类别归属频率与预测概率之间的相对熵。

最后一部分，里斯和克罗将多重序列比对的方法和加权方案相结合。为了避免使用序列谱时丢失窗中氨基酸的关联信息，首先对单一序列做出预测，然后再用多重序列比对的方法综合这些预测。这一策略在参考文献［587,457,358］中有涉及，可与任意一种比对算法相结合，适用于从一级结构中预测二级结构的所有方法。最终的预测是利用加权将多重序列比对的单一序列预测结果进行综合。为补偿数据库偏倚而采用的加权方案为最大熵值加权。[337]给定列中的单独打分值可用加权平均或权重优先的方式综合，这要看是对单一序列预测算法得到的概率值取平均值还是多选一。因为直至最终决策阶段，所有的信息仍被保留，所以加权平均性能较好。研究结果也证实了这种猜测，虽然加权平均和权重优先选择两种策略的差别不大。然后使用一个带有5节点单隐层的小型网络作为参考，应用多重序列比对方法对二级结构预测的结果进行过滤（详细内容见参考文献［445］）。该小型网络也可利用卷曲区域较不保守的性质，该性质使卷曲区域的每列在多重

序列比对时均获得较高的熵值。

大量的实验表明：（1）局部编码的网络结构可以避免过拟合问题；（2）使用诸如蛋白质标准化长度、蛋白质的氨基酸平均组成之类的系列附加信息作为额外输入，不会改善预测的结果；（3）对每个算法构成部分所引起的性能提高进行量化——例如多重序列比对使预测性能提高了5%，这主要来自对较保守的α、β结构的预测率的提高；（4）网络的输出可以理解为各类别归属概率。最重要的是，对罗斯特和桑德所使用的126种非同源蛋白质的数据库进行7倍率交叉验证实验时，得到的基本正确率为Q_3=66.3%。结合多重序列比对方法后，正确率达到Q_3=71.3%，相应的相关系数为C_α=0.59、C_β=0.50和C_c=0.41。这样，无论如何设计网络结构，最终性能实际上与参考文献［453］相一致。这无疑为使用局部信息进行预测的任何算法的准确率上限略大于70%~75%的论断提供了又一证据。

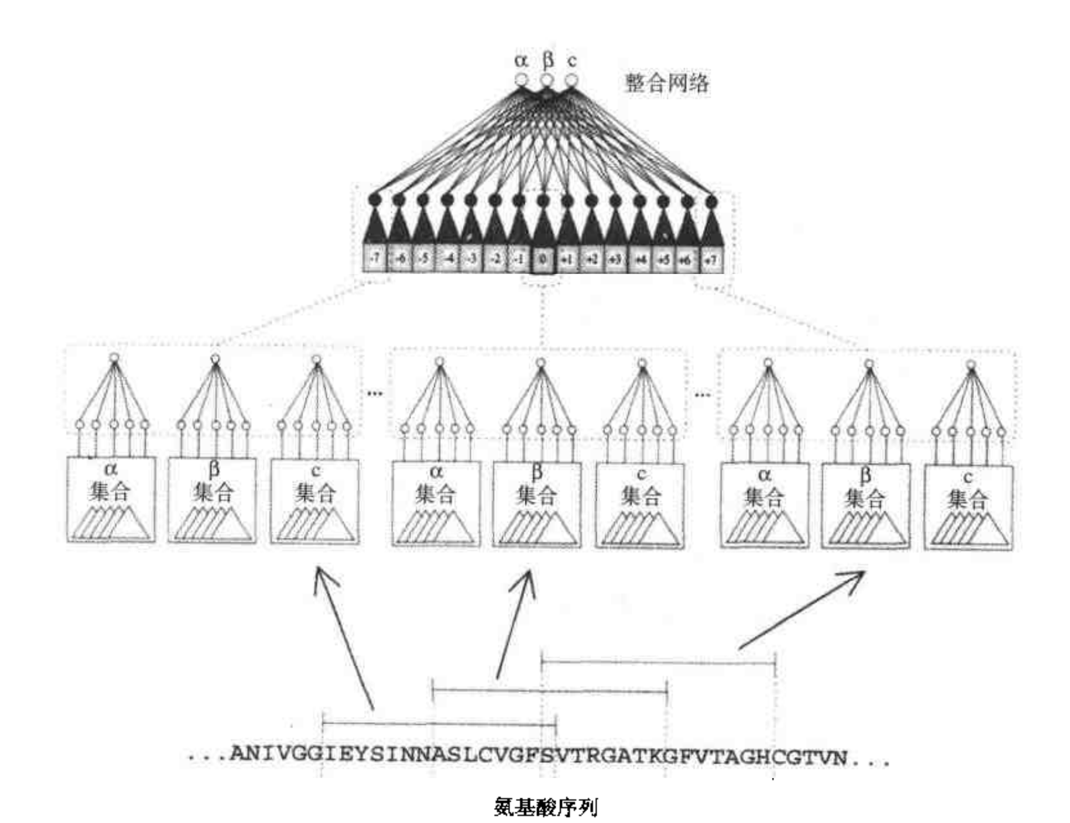


图6-4　里斯和克罗的网络整合、过滤预测算法

图上方的连接网络具有3×5×15的窗长，整合了图中间部分的二级结构预测结果。在网络中，对输入的三类结构的整合加权依赖于每个窗口的位置特异性。

6.3.4 基于位置特异性分值矩阵的二级结构预测

PHD方法的最大贡献在于使用了序列谱，其中包含更多的结构信息供神经网络提取。序列谱基于比对算法所找到的同源序列，而序列谱的质量显然依赖于寻找同源序列所使用的比对算法。

PSI-BLAST方法[12]采用了迭代搜索策略：先用一个单序列扫描数据库（如SWISS-PROT数据库）找到一组序列，这组序列又产生一组新的搜索序列谱，然后再用这组搜索序列谱寻找新的序列。这种“序列漫游”（sequence walking）策略通常能获得更多的序列家族成员，虽然它也可能导致选出非相关序列、减弱序列谱的结构保守性和家族特异性偏倚等不良后果。

PSI-PRED方法[290,386]中，约翰森（Jones）采用了这种迭代策略生成序列谱从而改良网络输入。这些序列谱基于位置特异性分值矩阵，大大提高了网络的预测性能。约翰森使用Blosum62替代矩阵进行数据库的初始扫描以获得序列谱，而在后继扫描过程中，替代矩阵由点对点的多重序列比对计算得到。重复若干次该过程，可得到网络的输入向量。

用这种更先进的扫描方法所得到的结果替代PHD方法中的HSSP序列谱，可以将三类别［螺旋（DSSP H/G/I）、折叠（DSSP E/B）和卷曲］的预测正确率提高到76.5%。若将G、I螺旋类型包括在卷曲类别中，预测准确率可进一步提高到78.3%。因此，依赖对二级结构的精确定义，预测正确率变化范围为1%~2%。这一变化范围与大多数神经网络方法以及其他方法保持一致。1998年举行的Asilomar CASP3（蛋白质结构预测技术评判）竞赛中，PSI-PRED方法在二级结构预测上被公认为是最好的。对于一个序列集合，其预测正确率可达77%；对于一个预测难度大的序列子集［该子集的规模相当于包括187个单一折叠子（fold）的大型测试序列］，其预测正确率可达73%。[324]

6.3.5 对800个不同的网络输出做平均的预测策略

正如在天然蛋白质中所观察到的，虽然折叠、螺旋对序列长度有一定要求，但这两种结构类型的长度分布域还是比较宽的。对于一个大型数据集，如果仅用单一网络进行预测，窗长的选择既要能够发现卷曲到非卷曲区域的过渡，又要能够发现非卷曲到卷曲区域的过渡，因此，窗长必须在两者中进行折中。大窗口可以利用二级结构中的附加信息，而小窗口识别小长度的结构更有效，这类小长度结构不与前后的二级结构发生重叠。单一网络中，窗口过大或过小都会降低预测性能。但对单独的一个样本而言，较大或较小的窗口往往能使识别更有效，也就

是说可以使输出值更接近于饱和值0或1。

当将许多不同的网络联合使用时，所面临的一大问题是如何从这些整体上次优，而实际上对局部数据更可靠的网络中获得收益。若网络的数量比较大，简单的平均会使源自各个次优网络的噪声累积，产生负面影响。有人认为，联合使用的网络数目的上限大约为8。[118] 在本项研究工作中，三类别预测的正确率大约为73.63%（1个网络）、74.70%（2个网络）、74.73%（4个网络）和74.76%（8个网络）。

但是彼得森（Petersen）及其合作者在最新的研究中指出：整合800个各不相同（不同的窗长、不同的隐层节点数目等）的网络，所得到的预测性能会提高。[427] 这一整合过程的关键在于识别800个网络中，哪些网络对测试集合中给定氨基酸残基的预测可靠性高。仅对高度可靠的预测结果做平均，使之能利用更多网络，避免引入次优网络的噪声，消除假性的正、负预测点。

使用这一策略可以将预测正确率提高，直到超过PSI-PRED方法的预测正确率。将DSSP分类的类别信息结合到螺旋、折叠和卷曲分类问题中后，氨基酸的预测正确率（对各氨基酸预测结果取平均）范围提高到77.2%（标准方式结合）和80.2%之间。蛋白质的单链平均的预测正确率也会有所提高（77.9%~80.6%）。

输出扩张

在这一研究中，通过在输出层引入另一新的特征，提高了预测的性能。彼得森提出了一种名为"输出扩张"（output expansion）的方法。利用该方法，网络不但预测输入窗中央的那个氨基酸的二级结构，而且同时预测邻近残基的结构。

这一思路与早期的思路有关，即通过训练隐含规则来构建网络结构，并利用隐含规则进一步限定网络权重，以利于提高网络模型的推广性能。

用于预测通货汇率（如美元兑换日元）的网络，如果必须同时用于预测美国预算赤字或其他与原始输出相关的特征，其预测结果可能会改善。[1] 这种思想也称为学习多相关任务或多任务学习。[115] 同时学习多个相关任务的网络，可以利用这些任务的相关信息作为推断参考，使学习过程更理想。

在蛋白质二级结构预测中，邻近残基的结构类别无疑与待预测残基相关。而其他隐含线索还包括残基的表面空间倾向（由PDB中的结构数据计算得到）或通过特定疏水性标度得到的残基疏水性。

6.4 信号肽及其剪切位点的预测

无论是在原核生物还是在真核生物中，信号肽都控制着几乎所有蛋白质到分

泌通路的“入口”。[542,207,440] 它们位于氨基酸序列的N端，在蛋白质转座到细胞膜时被剪切掉。

由于存在大量未经处理的数据，以及在重组系统中更有效地生产蛋白质的商业需要，信号肽及其剪切位点的自动识别引起人们的强烈兴趣。人们认为，在所有组织中，在许多不同种类的蛋白质中，指引蛋白质进入分泌通路的机制是相似的。[296] 但识别问题在某种程度上的确具有组织特异性。使用神经网络的预测方法将革兰氏阳性细菌、革兰氏阴性细菌以及真核生物区别对待，已证明能获得更显著的成效。[404,131] 源自不同蛋白质的信号肽，并不具有严格的保守序列——实际上，它们的序列相似性相当低。但是，它们共同拥有带7~15个疏水性氨基酸的中心结构（疏水核），蛋白质前体N端的带正电区域，以及在剪切位点前面的3~7个具有极性的（绝大多数不带电）氨基酸。

信号肽自动识别问题以及其他与“位点”相关的系列分析，一般有两条相互独立的解决思路：或者直接预测位点，或者将两种不同区域中的氨基酸划分为两种类别。在后面一条思路中，所有序列的氨基酸被分为剪切位点和非剪切位点。由于大多信号肽长度小于40个氨基酸，所以在分析中只考虑前60~80个氨基酸片断。当然，氨基酸也可按属于信号序列还是属于成熟蛋白序列加以分类。在下面的介绍中，这两种分类策略被结合使用，并互为补充信息。由于功能位点的预测是局域作用的，所以尽可能使用较小的窗口；而在区域功能匹配的预测中，为取得较好的预测效果，常需要较大的窗口。

6.4.1 SignalP预测程序

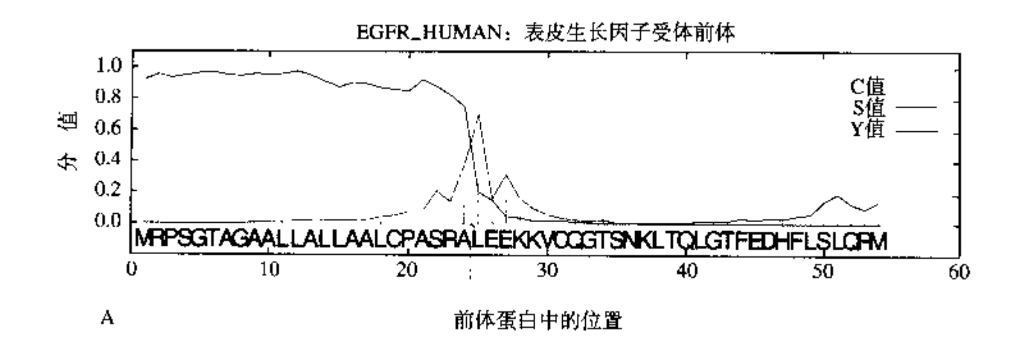
在SignalP预测程序中，[404] 所提及的两种类型的神经网络为序列中的每一个氨基酸设定了取值在［0, 1］的不同分值。S值，即信号肽/非信号肽神经网络的输出，可认为是对每个氨基酸属于信号肽的概率的估计；而C值，即剪切位点神经/非剪切位点神经网络的输出，可认为是对每个氨基酸是成熟蛋白质的第一个残基的概率的估计（值为+1的位置对应剪切位点所在）。

在图6-5中，给出了信号肽的C值和S值的两个例子。带有典型剪切位点的信号肽的分值曲线如图6-5A所示。其中C值曲线有一个尖峰，其位置对应于S值曲线的拐点位置。也就是说，对于C值及S值，该例子的识别结果都是正确的。另外一些非典型例子的曲线则形如6-5B所示，其中C值曲线呈多峰形态。

在该程序中，数据集被分为5个子集。依照交叉验证值，对每个预测问题以及各种组织类型，选择5个相互独立的神经网络。将这5个神经网络输出值做平均即可得到每个样本的C值和S值。这样，对于有三种组织类型的实际问题，每个分值都

包含了15个神经网络的输出信息。信号肽的识别，从另一个角度说明了对神经网络输出结果进行后期处理的重要性，以及使用好的编码方式可提高网络的整体性能。

使用非对称窗是解决C值问题的最好方法。在非对称窗中，剪切位点上游的氨基酸数多于下游的氨基酸数：上游取15个氨基酸，下游取2~4个氨基酸。这与剪切位点位置的模式信息（作为信号肽序列的标识）相对应。[404] 而使用对称窗，可以从整体上更好地解决S值问题，因为在识别信号肽/成熟蛋白的序列特征差别上，对称窗的性能显然更好。在人类序列中，窗长取27；在大肠杆菌的序列中，窗长取39。



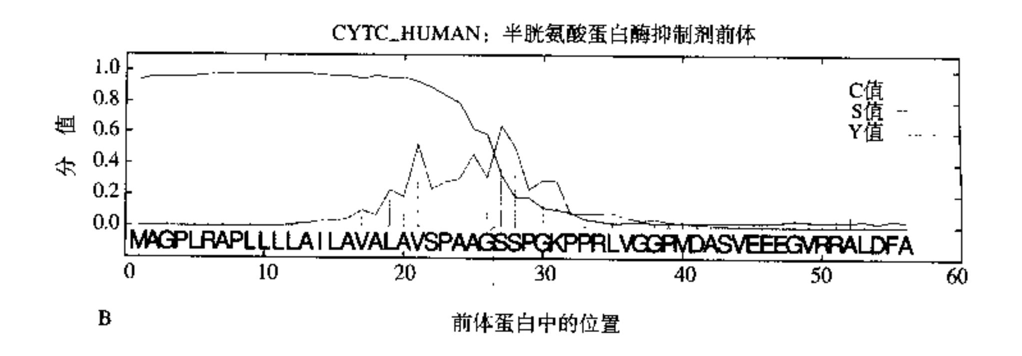


图6–5　带有可剪切信号肽的序列预测示例

图中显示了序列每个位置上的C值（剪切位点神经网络输出）值，S值（信号肽神经网络输出）值，Y值（S值和C值相结合的剪切位点打分）值，$Y_i=\sqrt{C_i \Delta_d S_i}$。C值和S值是用数据的不同子集训练的5个神经网络输出的平均。紧接在剪切位点后面的那个位置，即成熟蛋白质的第一个残基位置，C值较大。真实的剪切位点的位置用箭头标出。在图A中，使用C值或S值，序列的所有位置都被正确预测。在图B中，有两个位点的C值大于0.5。因此如果只依靠C值的最大取值原则，将无法预测出正确的剪切位点，而使用Y值就可以正确预测出剪切位点位置。

由于在大多数样本序列中，剪切位点只有一个，所以当每个位置用C值表示时，不必要求作为匹配标准的截断值（例如取0.5）是固定的。把C值取值最大的位置视做信号肽的剪切位点，并计算此时剪切位点被正确预测的序列数占被预测序列总数的百分比值，也可以在序列水平上评价C值神经网络的性能。早期的权重矩阵方法[539]也是如此计算性能指标。在序列水平上评估神经网络输出，使网络的性能有所提高。即使C值曲线无峰值或截断值上有多个峰值，仍可在C值最大的位置识别出真实的剪切位点。

如果C值的几个峰值强度相当，可以参照S值曲线识别出正确的剪切位点，因为C值曲线的峰值处恰恰对应信号肽区域与非信号肽区域的结合处。最好的方式就是取C值和S值的平滑差分的几何平均值作为综合的打分值，从而定义Y值：

$$Y_i = \sqrt{C_i \Delta_d S_i} \tag{6.2}$$

其中$\Delta_d S_i$是第i个位置的前d个位置的S值的平均值与第i个位置的后d个位置的S值的平均值的差：

$$\Delta_d S_i = \frac{1}{d}\left(\sum_{j=1}^{d} S_{i-j} - \sum_{j=0}^{d-1} S_{i+j}\right) \tag{6.3}$$

与C值相比，Y值提高了序列水平上的预测性能（百分比正确率），但没有提高对单独位点的预测性能（C_C）。图6–5B给出了C值预测错误而Y值预测正确的例子。

有趣的是，这种方法也适用于检测蛋氨酸转录起始位点的标注错误。对SWISS-PROT数据库中大量较长的信号肽的研究发现，在靠近N端5~15个氨基酸的序列位置常出现第二个蛋氨酸。[422] 图6–6显示了对人类高血压蛋白原序列的SignalP预测结果。在N端，序列的S值很低，但在序列的第二个蛋氨酸位置之后，S值增大到合理的水平。这一预测结果明显暗示该序列的转录起始点标注错误。

6.5 DNA/RNA序列分析的相关应用

6.5.1 遗传密码的结构和起源

自遗传密码规则首次提出后，[407] 为揭示遗传密码具有对称性[216,6,509,514,125]以及它的进化历史[168,563,294,569]，研究人员做了大量的尝试。在这类研究分析中，其中20个氨基酸的特性及其相似性发挥着重要的作用。从系统和纠错的角度来看，密码子匹配与氨基酸的物理特性相关联。三联体中的三个不同位置与氨基酸的不同

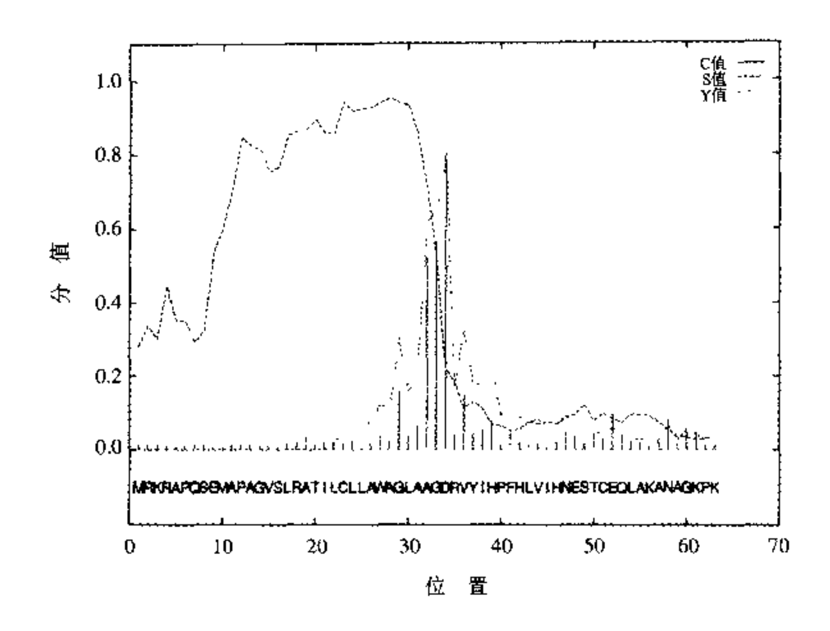


图6-6 人类高血压蛋白原（ANGT-HUMAN）序列的SignalP预测

在属于信号肽的残基位置上S值较大，而C和Y值在剪切位点的下一个位置——+1位点取得最大值。注意，S在第一个和第二个蛋氨酸之间的位置上取值较小。

特性相对应。第一个编码位置与氨基酸的生物合成途径相关，[569,514]也与“原始汤”（primordial soup）合成实验所揭示的进化相关。[159,478]第二个编码位置与氨基酸的疏水性相关。[140,566]退化的第三个位置则与分子重量或氨基酸的大小相关。[240,514]生物体通过两种方式利用这些特性进行编码纠错。其一是退化与蛋白质中氨基酸的数量多少相关，降低了随机突变导致氨基酸改变的几率。[371]其二是相似氨基酸的密码子相似，从而降低了突变导致蛋白质结构变异的几率。[140,216,6]

研究遗传密码结构的神经网络方法新颖而独特，整个分析方法不带偏倚，并完全由数据驱动。[524]神经网络方法直接依据密码和氨基酸之间的遗传密码标准匹配关系（图6-7）推断结构。因此，无需引入任何核苷酸或氨基酸之间的先验关系。

学习遗传密码的神经网络，其输入层为1个核苷酸三联体，输出层为相应编码的氨基酸。这样，输入层有64种三联体输入类型，输出层有20种氨基酸（如图6-8所示）。输入、输出是稀疏编码的，其中12个节点编码输入，20个节点编码输出。

带有3个或4个中介层节点的神经网络相对容易训练，而带有2个中介层节点的神经网络相对难训练得多。研究中采用自适应训练模式，才可能出现最小的神

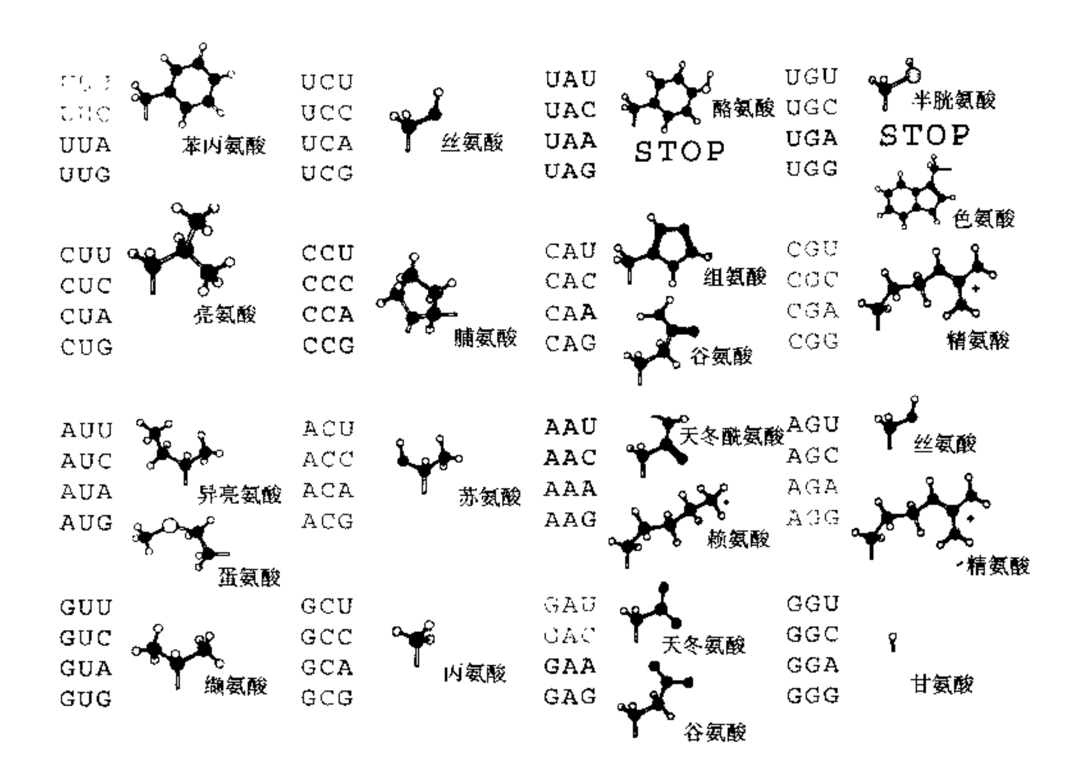


图6-7 标准遗传密码

编码同一氨基酸的多种三联体密码用同一灰度阴影显示。

经网络（带2个中介层节点）。[524] 研究发现，使用传统的反向传播算法训练的网络无法达到这样的最小规模。

训练前馈网络一般采用后向传播算法，其目标在于获得较低的网络误差$\mathcal{E}$，但分类误差$\mathcal{E}_C$并不一定很低。所以训练网络以获得$\mathcal{E}_C$=0.0的结果很困难。在一些文献中，提及了一些网络训练策略以获得较低的分类误差。一种简单有效的方法就是对易错分类的样本采用较高的学习率，而对易正确分类的样本采用较低的学习率。在最初的学习阶段，大多数样本容易错分，因此网络的学习率较高，从而大大降低了网络的误差$\mathcal{E}$。在后继的训练过程中，仅仅对难学习的样本采用较高的学习率，从而引入噪声使误差比较容易地过渡到较低的平稳态。

另一种有效的方法是改变不同类别样本的出现频率，从而使样本训练更为平衡。对于遗传密码，这意味着对应每种氨基酸，无论原始简并编码出现多少种密码子，输入神经网络的密码子的数量应该相同。因此，在训练集中，蛋氨酸密码子应该出现6次，而半胱氨酸密码子应该出现3次。这样，训练集的密码子数目就

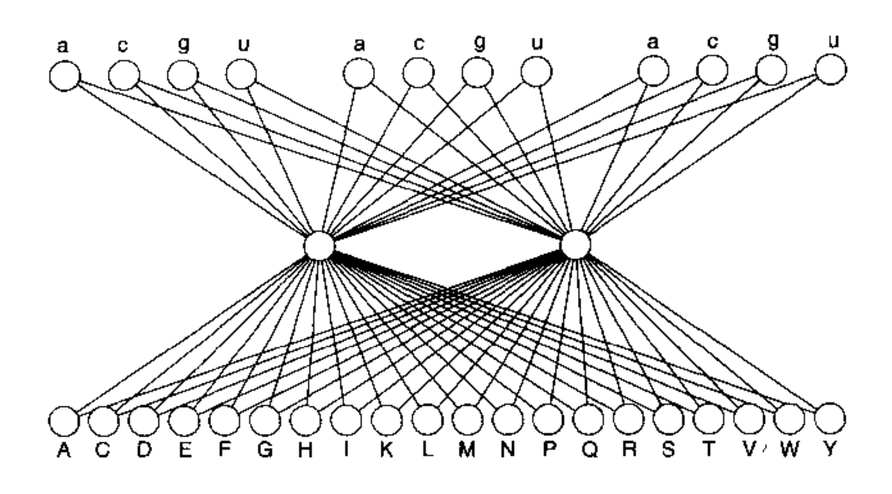


图6-8　学习标准遗传密码的神经网络结构

这个神经网络具有12个输入节点，2（或更多）个中介层节点和20个输出节点。输入层用3个4比特大小的二进制数字串来编码核苷酸三联体。对应腺嘌呤编码为0001，胞嘧啶编码为0010，鸟嘌呤编码为0100，尿嘧啶编码为1000。输出层用类似方法编码氨基酸，例如丙氨酸编码为10000000000000000000，半胱氨酸编码为01000000000000000000等。中介层和输出层节点的神经元激活值取0.0到1.0之间的实数。采用平衡学习的反向传播算法[456]来调整神经网络的参数（12×2+2×20=64个权重，2+20=22个阈值）。在这种算法中，对于每个密码子，训练循环次数与和该氨基酸相关的密码子数量成反比。因此，平均而言，蛋氨酸的训练循环数是亮氨酸的6倍。训练过程中，如果输出层节点的输出值比其他节点的输出都大（"胜者通吃"原则），就说明学习过程成功。每一训练周期中，密码子都是以随机顺序出现的。

由61个扩大为186个，每一个周期的训练时间也变为原来的3倍。

获得低分类误差的更为有效的方法是采用自适应的训练集，依据样本是否被现有的神经网络正确分类这一标准来决定是否保留这些训练样本。这一方法必将为学习过程引入更多噪声，以避免局部最小。通常在每个样本训练结束而不是在整个训练周期结束时更新网络参数，就能在训练过程中引入噪声。下一步就是在每一训练周期中，颠倒训练样本的次序。采用自适应方法，从样本集中顺序随机地挑选样本，更新网络参数，便不再存在训练周期的概念。为了增加不易学习的样本出现的频率，每个错分样本会被重新放人样本集中，用以替代其中的一个训练样本。为了确保样本没有遗漏，仅仅样本集合的一部分可以进行样本交换。经过足够的训练学习后，可以保证每个样本都被利用，并且不易学习的样本被用来训练网络的次数更多。综合而言，该过程如下：

1. 初始化训练样本集的第一部分和第二部分；
2. 从样本集中随机选择一个样本，将其输入到神经网络中；
3. 使用反向传播算法训练神经网络；

4. 如果该样本分类正确，则回到第二步继续进行；

5. 如果该样本分类错误，则将其放入样本集的第二部分，并随机替换其中的一个样本；

6. 重复该过程，直至E_C=0。

使用自适应的训练策略可以成功地训练带有2个隐层节点的神经网络。训练过程中，网络建立了遗传密码的内部结构。编码结构的内部表示由2个中介层节点的输出确定，很容易在平面上可视化。神经网络将61个表示氨基酸的编码密码子的12维向量映射为平面中的61个点。当网络学习成功时，20个输出节点便能够将这些点线性分开。

图6-9表示了神经网络通过自适应反向传播算法建立与遗传密码相对应的内部结构。每个密码子对应平面中以该氨基酸第一个字符表示的一点（x，y）。训练前，61个点在（0.5, 0.5）附近集中。在训练过程中，这些点以环形的轨迹彼此分离，最终分布在圆环边界上。

神经网络确定了3组密码子，分别对应于圆形区域的3个部分（见图6-9）。后来的研究发现，这3组密码子将由GES度量的转换自由能[166]划分为3个能量区间：［-3.7, -2.6］，［-2.0, 0.2］和［0.7, 12.3］（kcal/mol）（见表6-2）。惟一不符合这三类要求的是亲水性氨基酸——精氨酸，它是遗传密码中的一个例外。[319,509,514]编码精氨酸的密码子数量与自然界蛋白质中大量出现的精氨酸数量是矛盾的。[470]精氨酸在遗传密码中的地位很特殊。[294]螺旋结构中，它具有明显的与疏水性残基位于同侧的倾向性。[136]神经网络将精氨酸划入过渡类型，不属于3种类型。训练后的神经网络将3个终止密码子映射到与相似密码子相邻的位置（x，y）：UAA、UAG邻近Tyr（Y）；UGA邻近Trp（W）（图中未标出）。

神经网络至少需要2个中介层节点才能很好地学习遗传密码映射，这说明遗传密码本质上是非线性的。在分类问题中，这就意味着遗传密码是非线性可分的。这一事实适用于大多数研究人员使用的核苷酸的稀疏编码。对于以寻找核苷酸与氨基酸之间关系为目的的DNA或mRNA前体计算化分析问题，无论采用什么样的算法，都是个非线性问题。[100,102]很容易证明遗传密码确实是非线性的，因为丝氨酸的所有密码子无法用线性的方法与其他密码子分离。

与很多其他神经网络不同，这里所训练的网络的权重有一个较完善的结构（图6-10）。输入层到中介层的连接权重的大小反映了在密码子特定位置上的不同核苷酸的重要性程度。有趣的是，第二个密码子位置的权重最大，其余依次是第一位置和第三位置，这与早期的研究发现一致。[424]在很大程度上，两个中介层节点分担了分类功能。左边的节点受密码子第二个位置上的A或G碱基的影响很

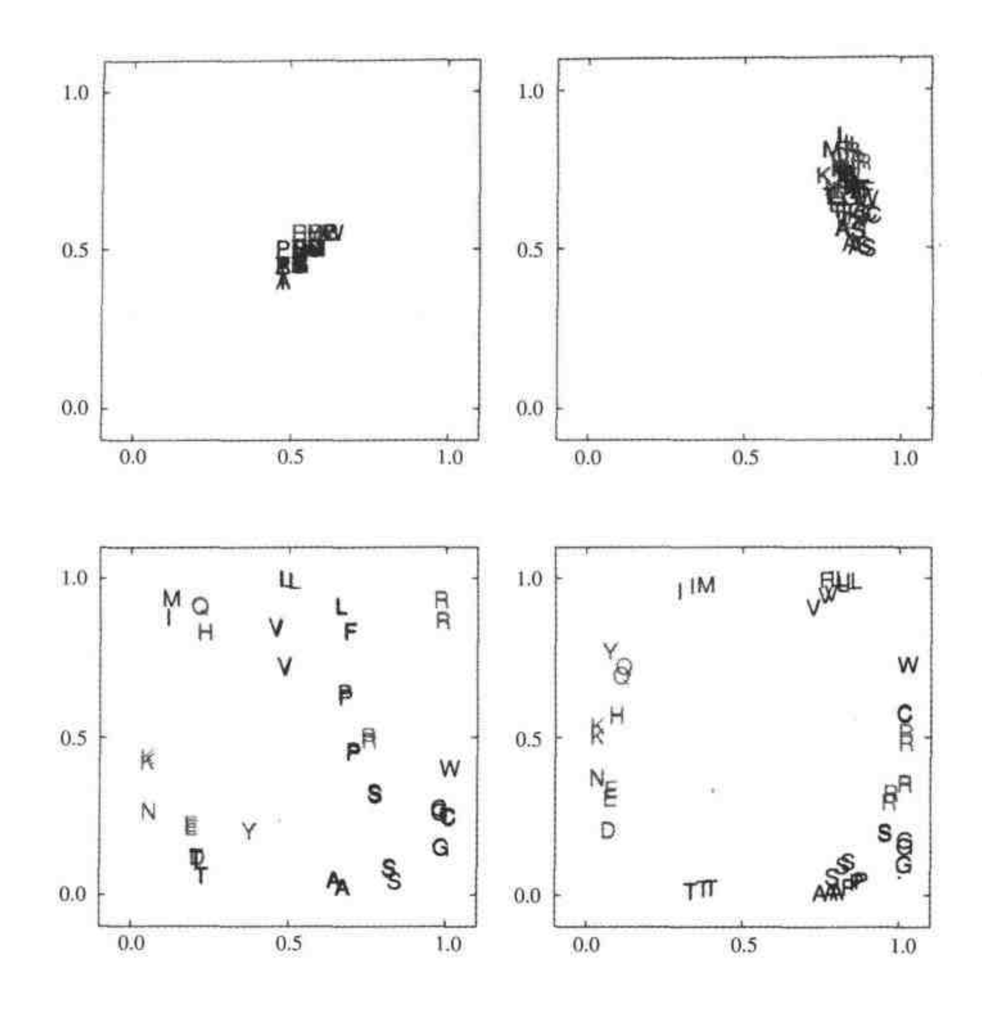


图6-9　遗传密码神经网络的隐层节点输出

每一点对应编码61种氨基酸的三联体的二维实值输出。随机赋予初始值的未训练网络中，61个点都分布在矩形的中心区域内。经过7个周期的训练后，这些点移动到瞬间局部最小化的位置，此时，隐层节点的输出趋于1，输出层节点的输出趋近于0。30个周期后，各点开始分离，各类型集合开始聚集但仍有混合。最终，13 000个周期之后，61个密码子分类聚集，散布在圆环区域上。对应上面的各子图，密码子被正确分类的数量依次为2、6、26和61。

大，右边的节点受密码子第二个位置上的C或U碱基的影响很大。同样，密码子第一个位置上的A和C，G和U分别影响着这两个节点。在遗传密码中，密码子第三个位置上的C和U对于所有氨基酸是等效的，而A和G对于大多数氨基酸（除Ile、Met和Trp外）也是等效的。神经网络通过对第三个位置上的两对密码子分别取正值和负值的权重来处理这类碱基等价性。

表6-2 氨基酸及它们以GES标度度量的转换自由能（kcal/mol）[166]

氨基酸	水-油	密码子
Phe	-3.7	UUU UUC
Met	-3.4	AUG
Ile	-3.1	AUU AUC AUA
Leu	-2.8	UUA UUG CUU CUC CUA CUG
Val	-2.6	GUU GUC GUA GUG
Cys	-2.0	UGU UGC
Trp	-1.9	UGG
Ala	-1.6	GCU GCC GCA GCG
Thr	-1.2	ACU ACC ACA ACG
Gly	-1.0	GGU GGC GGA GGG
Ser	-0.6	UCU UCC UCA UCG AGU AGC
Pro	0.2	CCU CCC CCA CCG
Tyr	0.7	UAU UAC
His	3.0	CAU CAC
Gln	4.1	CAA CAG
Asn	4.8	AAU AAC
Glu	8.2	GAA GAG
Lys	8.8	AAA AAG
Asp	9.2	GAU GAC
Arg	12.3	CGU CGC CGA CGG AGA AGG

GES度量值将位于三联体第二个碱基位置上对应为U、A的密码子与其他密码子分离，留下三联体第二个碱基位置上对应C、G的密码子作为中间类别对待。转换自由能是通过一个疏水特性计算项和两个亲水特性计算项得到的。前者由氨基酸的表面积大小决定，后者反映由于氢键作用而产生的极性影响和在pH=7的条件下将侧链转换为中性的能量需求。

密码子第二个碱基位置与氨基酸疏水性之间的关联关系非常基本，且具有一个显而易见的优点，即能够使疏水性氨基酸转变为亲水性氨基酸的变异或错翻译的可能性最小化。[538,409,87]早期的遗传密码研究中，密码子的类别是与氨基酸的类别相对应的。[562]这些类别大多数只与水环境下多肽链的折叠相联系。由于系统进化历史早于细胞质的脂膜，[316,73,117,76]在早期有关蛋白质器官合成的文献论述中，脂膜研究尚未显得十分重要。由于深受原始核糖体和基因是如何被附上脂膜结构这一问题困扰，细胞起源的研究无法深入。[117]布洛贝尔（Blobel）和卡维利亚-史密斯（Cavalier-Smith）的观点是：基因和核糖体具有类脂质体的泡状表面结构，从而引发了膜蛋白的插入机制。[73,117]因而，基于脂类环境中氨基酸的特性对遗传密码进行分类看来也是必要的。

6.5.2　真核基因的识别和内含子剪接位点的预测

从20世纪80年代初开始，针对新近测序出来的真核生物DNA，产生了各种各样用于蛋白质编码区域识别的方法。原则上有两种相互独立的方法可用于确定外显子区域：预测序列中供体（donor）和接纳体（acceptor）④位点交替序列的位置；或按照编码和非编码类别对核酸序列（或连续的核苷酸序列片断）进行分类。

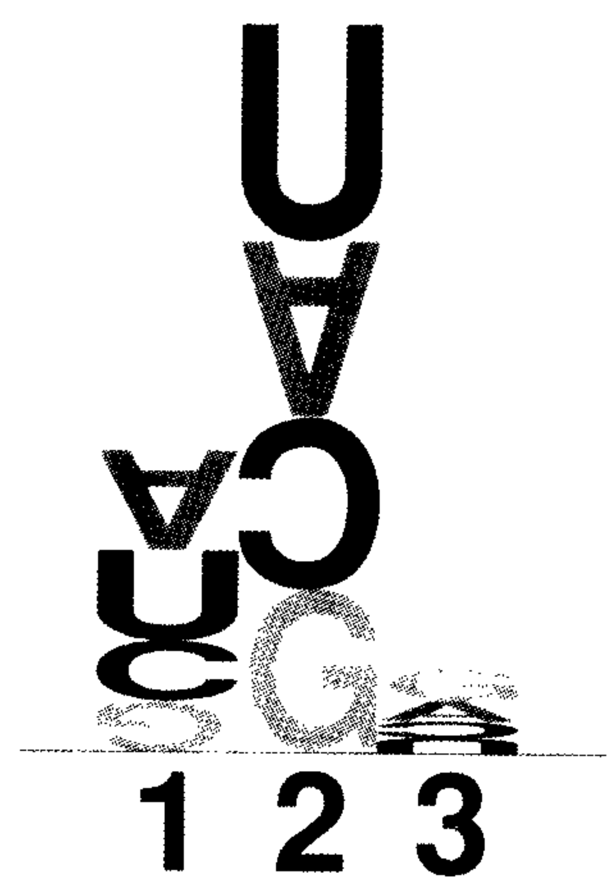


图6-10　训练后的遗传密码神经网络输入节点权重的图形化表示

对应三个碱基位置，字符的高度代表输入层节点到两个中介层节点的权重之和，如果和为负数，字符倒置。

④ "acceptor"在有些书中译为"受体"，为了避免与"receptor"（受体）混淆，本书统一采用《英汉生物化学及分子生物学词典》中的译法，即译为"接纳体"。　——译者注

内含子剪接位点区域在15~60个核苷酸的范围内，相对具有较保守的位置模式；编码蛋白质的区域（外显子）相对较长，一般为100~150个核苷酸，对于相当一部分真核生物而言，这一长度区间是比较稳定的。对于这两类位点，模式的长度或规律性直接影响了类别检测的正确率。

一些内含子剪接位点序列非常靠近序列空间中的“重力中心”，[344]而另外一些则相当偏离一般书中所描述的统一的序列模式［见图1-10中对阿布属拟南芥（*Arabidopsis thaliana*）供体位点的序列标识］。同样的，特定生物体的外显子序列也或多或少地与一般的阅读框模式相吻合。编码区域的模式保守性也与基因表达水平或蛋白质的氨基酸组成等相关。特定组织、特定基因上的编码蛋白质的密码子具有明显的三周期特性，这种周期性表现为在三个密码子位置上的有偏频率分布（biased frequency）。[525,305]在诸如细菌这样的物种中，这种频率倾向性在第一位置上最明显；而对于哺乳类生物，这种频率倾向性在第三位置上最明显（见图6-11）。富含脯氨酸、丝氨酸和精氨酸的蛋白质常常会产生阅读框错误，这与它们所包含的密码子在第一个和第二个位置上偏离标准模式有关。但在核糖体转录mRNA、确定编码阅读框的过程中，阅读框的功效是与密码子在三个位置上出现的概率有关的，而不是单单由密码子的平均使用频率决定。[525]图6-11显示了肠细菌（*Enterobacteria*）、哺乳动物、秀丽线虫和植物阿布属拟南芥的基因编码区域中，三联体编码各位置上的核苷酸分布偏倚。

在利用神经网络的方法预测内含子剪接位点的研究中，发现供体和接纳体位点序列模式和相关编码区域的序列模式间具有互补联系。[102]容易检测的外显子所对应的剪接位点模式不明显，反之亦然。尤其是作为编码区域标识的非常短的外显子，其标识信号弱而剪接位点模式却很明显。这种联系的强弱又受到各个物种间互不相同的内含子长度分布的影响。

在基于神经网络的预测算法NetGene[102]中，已经利用了剪接位点与外显子模式的互补关系，这种算法将两个局部剪接位点网络与一个窗长为301个核苷酸的外显子预测网络相结合。这一算法在相当程度上降低了预测的假阳性率。而且，在编码和非编码序列片断的过渡区域中，当外显子预测网络输出暗示该预测点可能为剪接位点时，这种算法可相应降低剪接位点网络的阈值，进而增强对较弱的剪接位点模式的预测性能（详见6.5.4节）。

6.5.3 综合多种特征预测基因结构

综合使用多个特征提取器来检测复杂对象的各种信号模式，在模式识别理论中应用已久。其中，一些基于神经网络结构的整合算法发挥了重要作用，如最早

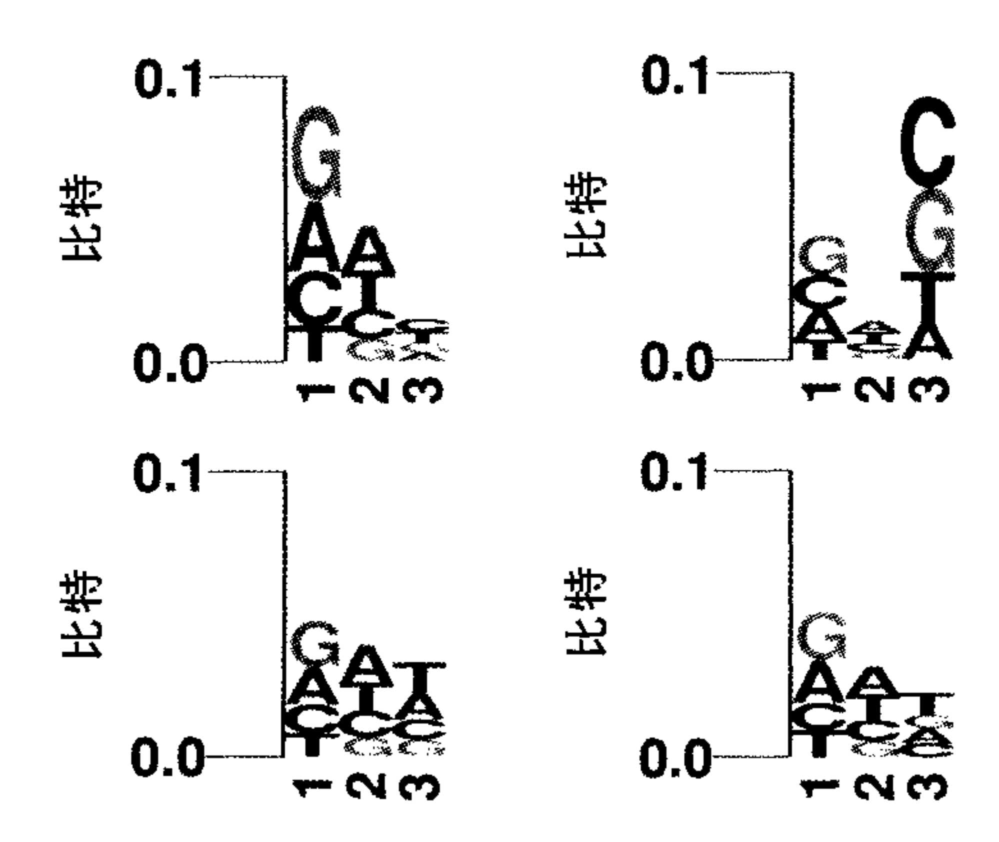


图6-11 四类不同物种的序列密码子分布

顶图分别是肠细菌和哺乳动物，底图分别是秀丽线虫和阿布属拟南芥。肠细菌的基因在密码子的第一个碱基位置上具有较强的倾向性，而哺乳动物在密码子的第三个碱基位置上的偏倚最明显。

的GRAIL和GeneParser程序。

GRAIL程序是基于神经网络的信号特征整合系统，用以识别编码区域。1991年，该程序首先将7种度量特征整合到一个预测网络中，从而更好地预测出编码区域。[528] 后来的GRAIL Ⅱ型程序可以对离散的编码区域进行预测，而不必像早期程序那样采用固定长度的滑动窗来预测编码区域。[529] GRAIL Ⅱ型程序可以将待预测的编码区域的长度作为网络的一个输入特征，从而可以针对较长或较短的外显子修改受长度影响的其他特征度量值。

多年来，GRAIL程序的性能不断提高，其原因在于信号特征检测的不断改良，而不在于神经网络的不断复杂化。而进行比较时，网络结构一直使用单隐层的MLP，利用反向传播学习算法进行训练。其中一种性能最好的特征提取器，是采用15阶非齐次马尔可夫链对DNA中的6元组进行特征提取。[529] GRAIL程序不但

可用于编码区域识别，也可用于基因建模（外显子组装），模型误差的检验和修正，CpG岛检测及PolII启动子和多聚腺苷酸位点的识别。

GeneParser程序[494]通过神经网络对外显子/内含子和剪接位点模式特征进行加权，估计序列片断属于不同类别（前端的、中间的或后端的外显子或内含子）的对数似然度。然后可利用这些数值，使用动态规划算法，寻找外显子和内含子的组合，使得似然函数最大。用这种方法，可以很快地获得多个次优解。每个次优解表征一处外显子—内含子的转换接合。研究人员还采用碱基部分替代和阅读框平移的手段引入噪声，研究该方法的鲁棒性，表明程序是如何在误差允许的范围内保持较好的预测性能的。

动态规划（DP）算法可用于精确识别基因组DNA序列中的内部外显子和内含子。GeneParser程序首先根据剪接位点存在可能性和外显子、内含子特异性度量值（包括密码子使用频率、局部组成复杂性、六元组频率、长度分布、周期不对称性等）来为序列打分，然后将这些信息组织起来供动态规划算法处理。GeneParser程序引入动态规划算法时，要满足外显子、内含子必须相邻且无重合的约束，并从中找出满足约束条件的综合评价分数最高的外显子、内含子组合。各分类过程的网络权重是由训练简单的反馈神经网络使预测正确率最大而得到的。研究人员曾经尝试在总共158 691个碱基的人类基因序列中，使用包含150个内部外显子的56个基因片断来训练该系统。在使用训练数据对网络进行测试时，GeneParser程序对外显子的识别率为75%，对编码核苷酸的识别率为86%，仅有13%的非外显子碱基被预测为编码碱基。对应外显子识别的相关系数值为0.85。由于网络权重学习算法较为简易，网络对新的样本数据几乎具有同样好的推广性能。

6.5.4 结合局部和全局信息预测内含子剪接位点

使用编码/非编码区域预测与剪接位点预测相结合的NetGene预测算法进行研究时，研究人员发现了剪接位点模式与编码区域模式的互补依赖关系。[102] 1991年首次开发的NetGene程序仅仅用于训练处理人类序列。1992年后，这个程序开始具有因特网支持功能（netgene@cbs.dtu.dk）。这一方法将三个独立的网络联合在一起：一个用来预测编码/非编码区域的全局神经网络调整着两个局部的供体和接纳体位点预测网络的匹配阈值。三个网络的窗长依次为301、15和41bp。用从外显子到内含子区域过渡时的全局信号突变调整局部剪接位点预测网络的实际阈值，而不是使用固定阈值。以上做法的目的在于提高供体、接纳体位点的正确预测比率。在外显子预测网络输出突然下降的区域，供体位点出现的可能性应该增

大，接纳体位点出现的可能性应该减小。而在外显子预测网络输出突然上升的区域，情况正相反。在外显子预测网络输出变化不明显——比如状态值持续为高（处于外显子区域中）或状态值持续为低（处于内含子区域、非转录外显子区域和基因间的DNA区域中）——的区域，为了降低假阳性率，需要提高相应剪接位点预测网络的预测可信度。

实际操作中，用给定位置右端序列的网络输出之和减去左端序列的网络输出之和，再用该差值除以输出节点的总个数，所得值作为输出层神经元输出的差分值，可以预测编码/非编码区域的转换边界。为了减少计算中同时使用的外显子和内含子区域位置点的个数，求和范围的大小设定为75个碱基——训练集中内部外显子的长度平均值的一半，从而使内含子的3'末端的输出差分值趋近于+1，而5'末端的输出差分值趋近于-1，以便于更好地检测编码/非编码区域。

图6-12中给出了测试集中的编码/非编码区域信号的平均特征值，差分值Δ和输出值超过0.25的供体、接纳体位点的信号值，这些数值取自GenBank的HUMOPS序列。[102] 注意处于内含子区域和序列的非转录区域中的一些局部也会显示出类似外显子的输出值特性。

在剪接位点预测网络输出阈值可调整的算法中，常用下列表达式评估外显子网络输出值占整个独立剪接位点预测网络的输出O的比重：如果

$$O_{供体} > e_D\Delta + c_D \tag{6.4}$$

则识别结果为剪接供体信号；如果

$$O_{接纳体} > e_A\Delta + c_A \tag{6.5}$$

则识别结果为剪切接纳体信号。其中差分值Δ的计算方法如上。常数c_D和c_A等于通常的网络输出截断阈值（cutoff），而e_D和e_A控制着外显子预测网络输出对最终预测结果的影响程度。这四个参数反映了供体/接纳体位点识别网络与编码/非编码区域预测网络间的互补侧重关系（relative strength）。

这四个合适的参数值决定了剪接位点识别的正确率和相关系数。[102] 与其他方法比较，使用外显子预测网络输出来控制截断阈值时，根据不同的预测精度需求，假阳性率可降低至1/2到1/30。

编码/非编码区域预测和剪接位点预测由输出神经元的输出水平决定，具有优势互补的特性。一般来说，长度小于75bp的外显子，其预测网络输出神经元的输出水平较低，为0.3~0.6；而供体、接纳体位点预测网络输出神经元的输出水平较高，为0.7~1.0。反之，对于较长的外显子，其预测网络输出的峰值尖锐程度较明

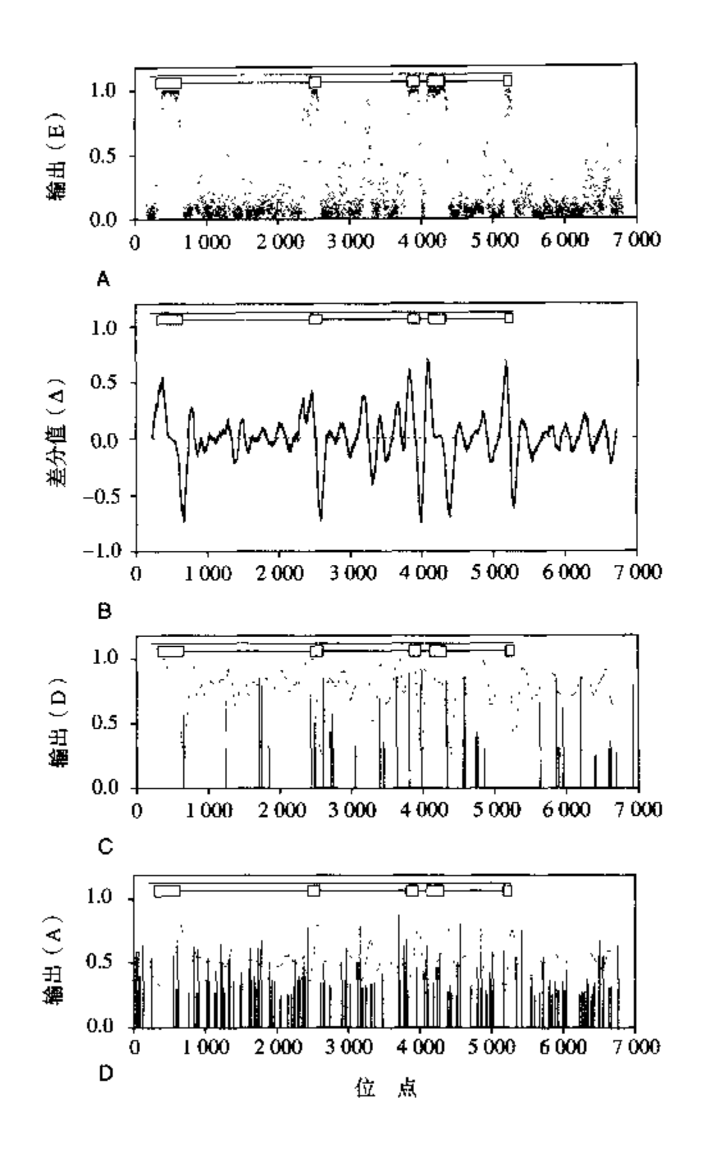


图6−12 使用NetGene方法预测GenBank中著名的HUMOPS序列剪接位点的操作步骤图

A. 编码/非编码区域预测网络的神经元输出，较强的信号对应正确的外显子区域。在内含子区域和序列终端的非转录区域，一些部分显示出类似外显子区域的输出值特性。小方框对应正确的外显子区域，连线对应内含子区域，而最顶端的直线表示DNA的整个转录序列。B. A图输出的差分值。C. 供体位点预测网络输出不小于0.25的供体位点输出值。D. 接纳体位点预测网络输出不小于0.25的接纳体位点输出值。序列上各位点的可变截断阈值（对应90%的真实剪接位点识别率）连成的曲线如虚线所示。

显，而供体、接纳体预测网络的输出模式不明显。

类似的NetPlantGene程序用于预测阿布属拟南芥的剪接位点。[245] 该生物是

第一种被完全测序的植物。与许多其他植物相比（见图1-2），其基因组长度（400Mbp）比较适中。

6.5.5　通过监测神经网络学习过程进行序列分析

众所周知，神经网络具有很好的样本推广能力。近来的研究发现，神经网络还具有另一种特性：监控其学习过程，可获得关于所训练样本对象内部结构的重要信息。神经网络并非以随机顺序进行样本学习，而是依赖可调参数的数量，先学习数据中线性可分的部分，然后再学习不属于主流特征模式的野点（outlier）。钱和塞诺斯基的早期工作[437]和其他预测蛋白质螺旋结构的工作[244]都清晰地说明了这一点。训练过程分为两步：线性可分的那部分数据学习速度快，而剩下的样本被神经网络正确分类的速度要慢得多。前一步中被学习的一些样本在后一步的学习中通常是不需要的。正如前面所叙述的关于遗传密码的研究一样，一个类似但更为复杂的研究分析有待进行。

神经网络学习一组样本的顺序揭示了每个样本的相对非线性信息，以及全数据集的规律性。[97,98] 这反过来可用于识别与主流模式不一致的异常样本，这些异常样本是使用了不合适的分类策略，或者仅仅由于随机导入的分类误差造成的。构建复杂的系统模型和避免错误规则是一对矛盾，神经网络具有很好的平衡这对矛盾的能力，因此可以处理噪声较大的数据，从中获得较高质量的修正反馈信息。

不论在序列分析领域还是在其他存在输入样本数据类别错误的问题中，神经网络的样本纠错能力在很多不同的工作中得到运用。利用神经网络预测内含子剪接位点的方法，提到了如何过滤不同来源的误差噪声。[100,101] 在训练过程中，有两种不同的方式可以用来监控样本的学习：一种将训练集视做一个整体，另一种则是独立分析每个输入样本是否被成功学习。将训练集视做一个整体的策略是通过观察整个网络误差$\mathcal{E}$下降的情况来衡量网络性能的。若$\mathcal{E}$相对大量的训练集样本保持恒定，则说明进一步的训练不会再提高网络的性能。在另一种策略中，如果样本的网络实值输出与目标输出皆处于截断阈值的同侧，则表明该输入样本已成功完成学习。划分两类输出类别的截断阈值绝大多数取值为0.5。这样，在网络训练过程的任意时刻，是否学习成功的评判标准都是针对未被正确分类的那些输入样本的。

表6-3显示了有限样本资源的小型网络进行供体位点预测的学习过程。大多数的样本学习速度很快，而剩下的那些则需要经历好几个循环周期。通过检查不能被学习的单样本输入，可以发现大量虚假的供体位点。这些虚假位点由于未被

充分加以识别认证，实验误差或者对实验结果的错误解释而混入公共数据库中。在监控训练过程中，可以获得有关供体位点输入样本的规则模式信息。例如对于供体位点监测问题，与较晚学习的输入样本相比，训练过程中较早学习的输入样本显示出较一致的5'端 $\frac{C}{A}$AG/GT$\frac{G}{A}$AGT序列保守模式。

表6-3　训练过程中，较早和较晚学习的供体位点输入样本序列

训练周期	GenBank的序列名称	序　列
1	HUMA1ATP	TACATCTTCTTTAAAGGTAAGGTTGCTCAACCA
1	HUMA1ATP	CCTGAAGCTCTCCAAGGTGAGATCACCCTGACG
1	HUMACCYBA	CCACACCCGCCGCCAGGTAAGCCCGGCCAGCCG
1	HUMACCYBA	CGAGAAGATGACCCAGGTGAGTGGCCCGCTACC
1	HUMACTGA	GCGCCCCAGACACCAGGTGAGTGGATGGCGCCG
1	HUMACTGA	AGAGAAGATGACTCAGGTGAGGCTCGGCCGACG
1	HUMACTGA	CACCATGAAGATCAAGGTGAGTCGAGGGGTTGG
1	HUMADAG	TCTTATACTATGGCAGGTAAGTCCATACAGAAG
1	HUMALPHA	CGTGGCTCTGTCCAAGGTAAGTGCTGGGCTACC
1	HUMALPI	CCTGGCTCTGTCCAAGGTAAGGGCTGGGCCACC
1	HUMALPPD	TGTGGCTCTGTCCAAGGTAAGTGCTGGGCTACC
1	HUMAPRTA	CCTGGAGTACGGGAAGGTAAGAGGGCTGGGGTG
1	HUMCAPG	GAAGGCTGCCTTCAAGGTAAGGCATGGGCATTG
1	HUMCFVⅡ	GGAGTGTCCATGGCAGGTAAGGCTTCCCCTGGC
1	HUMCP21OH	CACCTTGGGCTGCAAGGTGAGAGGCTGATCTCG
1	HUMCP21OHC	CACCTTGGGCTGCAAGGTGAGAGGCTGATCTCG
1	HUMCS1	GTGGCAATGGCTCCAGGTAAGCGCCCCTAAAAT
1	HUMCSFGMA	AATGTTTGACCTCCAGGTAAGATGCTTCTCTCT
1	HUMCSPB	AAAGACTTCCTTTAAGGTAAGACTATGCACCTG
1	HUMCYC1A	GCTACGGACACCTCAGGTGAGCGCTGGGCCGGG
…	…	…
2	HUMA1ATP	CCTGGGACAGTGAATCGTAAGTATGCCTTTCAC
2	HUMA1ATP	AAAATGAAGACAGAAGGTGATTCCCCAACCTGA
2	HUMA1GLY2	CGCCACCCTGGACCGGGTGAGTGCCTGGGCTAG
2	HUMA1GLY2	GAGAGTACCAGACCCGGTGAGAGCCCCCATTCC
2	HUMA1GLY2	ACCGTCTCCAGATACGGTGAGGGCCAGCCCTCA
2	HUMA1GLY2	GGGCTGTCTTTCTATGGTAGGCATGCTTAGCAG
2	HUMA1GLY2	CACCGACTGGAAAAAGGTAAACGCAAGGGATTG
2	HUMACCYBA	GCGCCCCAGGCACCAGGTAGGGGAGCTGGCTGG
2	HUMACCYBA	CAGCCTTCCTTCCTGGGTGAGTGGAGACTGTCT
2	HUMACCYBA	CACAATGAAGATCAAGGTGGGTGTCTTTCCTGC
2	HUMACTGA	TCGCGTTTCTCTGCCGGTGAGCGCCCCGCCCCG
2	HUMADAG	CTTCGACAAGCCCAAAGTGAGCGCGCGCGGGGG

（续表）

训练周期	GenBank的序列名称	序　列
2	HUMADAG	TGTCCAGGCCTACCAGGTGGGTCCTGTGAGAAG
2	HUMADAG	CGAAGTAGTAAAAGAGGTGAGGGCCTGGGCTGG
…	…	…
11	HUMCS1	AACGCAACAGAAATCCGTGAGTGGATGCCGTCT
11	HUMGHN	AACACAACAGAAATCCGTGAGTGGATGCCTTCT
52	HUMHSP90B	CTCTAATGCTTCTGATGTAGGTGCTCTGGTTTC
80	HUMMETIF1	ACCTCCTGCAAGAAGAGTGAGTGTGAGGCCATC
112	HUMHSP90B	ATACCAGAGTATCTCAGTGAGTATCTCCTTGGC
113	HUMHST	GCGGACACCCGCGACAGTGAGTGGCGCGGCCAG
113	HUMLACTA	GACATCTCCTGTGACAGTGAGTAGCCCCTATAA
151	HUMKAL2	ATCGAACCAGAGGAGTGTACGCCTGGGCCAGAT
157	HUMCS1	CACCTACCAGGAGTTTGTAAGTTCTTGGGGAAT
157	HUMGHN	CACCTACCAGGAGTTTGTAAGCTCTTGGGGAAT
164	HUMALPHA	CAACATGGACATTGATGTGCGACCCCCGGGCCA
622	HUMCFVⅡ	CTGATCGCGGTGCTGGGTGGGTACCACTCTCCC
636	HUMADAG	CCTGGAACCAGGCTGAGTGAGTGATGGGCCTGG
895	HUMAPOCIB	TCCAGCAAGGATTCAGGTTGTTGAGTGCTTGGG
970	HUMALPHA	CGGGCCAAGAAAGCAGGTGGAGCTGGGGCCCGG
2 114	HUMAPRTA	ATCGACTACATCGCAGGCGAGTGCCAGTGGCCG

所使用的网络规模很小（窗长为9，2个隐层节点和1个输出层节点）。训练样本是长度为33bp的序列片断，这些序列片断属于数据集的第1部分，位于331个剪接位点周围。表中显示了每个周期中能被训练的神经网络正确预测的以碱基G为中心的供体位点周围的碱基序列。与标准供体位点序列的保守模式 $\frac{C}{A}$AG/GT$\frac{G}{A}$AGT 相差较大的序列片断，在多个周期以后才被成功学习。

6.6　预测的性能评价

多年的研究积累了多种用来评价特定预测算法准确率的方法。[31] 对一些预测算法进行优化，是为了得到更低的假阳性率；而对另一些算法进行优化，是为了得到更低的假阴性率。一般来说，无论对于什么类型的预测算法，其目的是为了保证这些算法针对一些在构建算法的过程中未曾出现的新数据，同样具有很好的预测性能。也就是说，该预测算法应该能够对属于同一数据域的新样本具有推广性能。

从不同的层次衡量预测正确率是有意义的。例如，在信号肽预测中，预测正确率除了可以由正确预测属于信号肽的残基个数衡量外，还可以由计算被正确分类为信号肽或非分泌性蛋白的序列的个数衡量。类似地，蛋白质二级结构的预测性能也可从每条链或每个氨基酸的不同角度评价。

着眼点越高，预测性能的度量就越复杂，问题的针对性也越强。例如在信号肽预测中，统计剪切位点被正确预测的信号肽序列的数量也是有意义的。识别基因时，外显子两端的序列可能预测得完全正确，或仅仅产生一定程度的重叠。博塞特（Burset）和古哥（Guigo）[110]着眼于外显子层次，定义了4种简单的基因识别正确率的度量——灵敏度、特异度、“丢失外显子”、“错误外显子”——对完全正确或完全错误的预测结果进行统计。二级结构预测中，由于二级结构单元（螺旋或折叠）没有精确的定义，上述度量方法显然太粗糙了。此时可以应用片断重叠指标（SOV）来衡量算法的性能。[454,580]这是一套基于序列片断的启发式评价指标，其中正确预测的序列片断分值最大，即使预测结果与整个片断的类别并不完全一致。这一打分值尽量避免了在同一类别的序列片断中出现其他预测类别这一情况的发生，例如两个螺旋预测片断，其中一个片断相对另一个片断，连续的螺旋结构预测区域比较短，其对应的惩罚打分值（score punish）必然比较高。这一度量策略很好地反映了各种结构类型的片断的边界具有不确定性。这一例子说明，当更多地考虑预测问题的精确度时，高级别的正确率度量也将随之出现并更加有针对性（ad hoc）。

为了提高算法的推广性能，可集中考虑单残基/核苷酸层次的评价度量标准。在二级结构预测中，考察长度为N的氨基酸序列，相应各位置点的预测目标为$\mathbf{D}=d_1,\cdots, d_N$。简化起见，先考虑两类别分类问题，如$\alpha$螺旋/非$\alpha$螺旋。这样，$d_i$取值一般为0或1。当然，如果$d_i$表示氨基酸外表面积，或者是相应位置的概率或置信度（这个概率或置信度反映了现有知识的不确定程度）时，也可在［0, 1］上取值。对于多类别问题，如三类别（α螺旋、β折叠和卷曲）也是类似的。先假设预测算法或模型的预测输出为$\mathbf{M}=m_1,\cdots, m_N$。一般地，$m_i$为反映预测置信度的取值在［0, 1］上的概率。当然也可以利用阈值截断或“胜者通吃”原则获得离散的输出——0或1。于是面临的最基础的和一般化的问题是：如何评价$\mathbf{M}$的正确性，或者说如何比较$\mathbf{M}$和$\mathbf{D}$？

在不同时期，针对不同研究内容，研究人员提出了各种各样的方法，这势必会在一定程度上引起混乱。预测正确率与每种类别个体的出现频率强烈相关。在蛋白质二级结构预测中，天然蛋白质中非螺旋类别几乎占70%，螺旋类别仅占30%。因此，把所有个体都预测为非螺旋，也能获得70%的预测正确率，但这无疑不包含任何有用的信息。

下面回顾一下几种方法，并考察它们之间的联系和各自的优缺点。

所有这些方法都基于“氨基酸各位置点是独立和等价的”这一简单的基本假设。基于此，我们可假设：N、C端附近位置的残基对预测结果的影响与其

他位置的影响相同，不随权重学习算法而改变。我们还可以假设：无任何内在机制可以保证局部预测结果在某种程度上是“光滑”的（即若一个残基属于α螺旋，并不说明周围的若干残基属于α螺旋的可能性会增大）。相反地，在预测诸如内含子剪接位点、转录起始位点、糖基化或磷酸化位点等功能位点时，假设预测结果不是正确就是错误，那么该预测结果对几乎确信的位点就没有任何意义。

在独立和等价的假设下，如果**D**和**M**取值均为二值的，则算法性能的优劣可完全由下面的四个数值概括：

- TP=d_i为螺旋，m_i为螺旋的样本数（真阳性）
- TN=d_i为非螺旋，m_i为非螺旋的样本数（真阴性）
- FP=d_i为非螺旋，m_i为螺旋的样本数（假阳性）
- FN=d_i为螺旋，m_i为非螺旋的样本数（假阴性）

并且满足$TP+TN+FP+FN=N$。而**D**或**M**不是二值时，情况将更为复杂，再不能用这四个数来完全评价算法的性能。当**M**取值不是二值时，采用阈值截断的方法，仍可获得二值的预测结果。TP、TN、FP和FN值将会随着阈值选择的不同而变化。可将TP、TN、FP和FN值排入一个2×2的矩阵中：

	M	$\overline{\mathbf{M}}$
D	TP	FN
$\overline{\mathbf{D}}$	FP	TN

单独使用这四个值，尚不能显而易见地显示出给定方法的性能优劣。所以很多算法倾向于建立一个单一指标，以表示**D**、**M**之间的“距离”。但是必须清楚一点，即从四个值中归纳出一个单一的指标显然会丢失部分信息，即使在二值问题中也是如此。通常，由不同的TP、TN、FP和FN值可以衍生出相同的距离。下面，我们将给出一些关于**M**的性能度量函数，并比较分析它们的优缺点。

6.7 不同的性能评价标准

6.7.1 百分比

第一种显而易见的方法是用TP、TN、FP和FN计算百分比。例如周（Chou）和法斯曼（Fasman）[128,129]计算螺旋结构预测正确的样本所占的百分比：

$$PCP(\mathbf{D},\mathbf{M}) = 100\frac{TP}{TP+FN} \tag{6.6}$$

这与6.7.9节中敏感度的表达公式相同。单独使用这一指标不能提供任何有关假阳性的信息。假阳性信息可由非螺旋结构的预测正确率获得：

$$PCN(\mathbf{D},\mathbf{M}) = 100\frac{TN}{TN+FP} \tag{6.7}$$

文献［128,129］中使用了前面两个数值的平均值，称做Q_α。虽然Q_α是一项很有用的指标，但也容易使人产生误解，[549]并且仅在**D**和**M**为均二值时方可计算得到。直觉上，任何由TP、TN、FP和FN中的两个数值构造的数值指标在某种程度上都具有相当的偏倚。比如，一组（TP，TN，FP，FN）和另一组（TP，TN'，FP，FN'），无论其实质如何不同，取TP、FP计算所得到的指标值都是一样的。

6.7.2 汉明距离

在两类别分类的二值例子中，**D**与**M**间的汉明距离（Hamming distance）定义为

$$HD(\mathbf{D},\mathbf{M}) = \sum_i |d_i - m_i| \tag{6.8}$$

显然该式的结果为假阳性样本和假阴性样本之和$FP+FN$。这与单一的百分比度量等效。该距离没有将属于给定类别的样本比例计算在内。样本比例与50%相差得越远，这种度量越不具有代表意义。在非二值实例中，汉明距离被称做L^1距离。

6.7.3 二次距离

二次距离也称做欧几里德距离或LMS（最小均方差）距离，它的定义如下：

$$Q(\mathbf{D},\mathbf{M}) = (\mathbf{D}-\mathbf{M})^2 = \sum_i (d_i - m_i)^2 \tag{6.9}$$

严格地说，上面定义的距离应该取平方根值（参见下一节中的L^2距离）。在纯粹的二值问题中，二次距离退化为汉明距离，也等于$FP+FN$。该距离有利于定义非二值变量，常常在高斯模型的负对数似然度的定义式中出现。

$$P(d_i|m_i)=\frac{1}{\sigma\sqrt{2\pi}}\exp\left[-(d_i-m_i)^2/2\sigma^2\right] \tag{6.10}$$

其中σ为与$Q(\mathbf{D},\mathbf{M})$相关的缩放比例因子。对应二值变量，二次距离等效于汉明距离。但其主要的缺点在于高斯模型通常不适用于预测问题，二次距离值也无法真实地反映样本相对指定类别的位置比例。另一问题是，由于m_i和d_i取值在0和1之间，所以LMS距离的动态范围有限。这对于需要使用较大的误差信号值加速学习过程的学习算法并不理想。为避免这一问题，可以对LMS距离取对数值，得到

$$LQ(\mathbf{D},\mathbf{M})=-\sum_i\log\left[1-(d_i-m_i)^2\right] \tag{6.11}$$

这种改良后的误差函数已经被用于很多神经网络的学习中，见参考文献［99, 245, 236］。

6.7.4 L^P距离

更一般地，L^P距离定义为

$$LP(\mathbf{D},\mathbf{M})=\left[\sum_i|d_i-m_i|^p\right]^{1/p} \tag{6.12}$$

这一距离适用于任何类型的数值变量。当$p=1$，对应汉明距离；$p=2$，对应欧几里德距离。当$p\to\infty$时，L^∞距离变为一个超距离：$\max_i|d_i-m_i|$。这个距离提供了最差情况下的一个上限，但对于评价蛋白质二级结构的预测性能没有什么帮助。p的其他取值在实际中很少使用，对于评价预测性能也没有帮助。在二值问题中，L^P距离退化为（$FP+FN$）$^{1/p}$的形式。当$p=1$时，L^p退化为汉明距离。

6.7.5 相关系数

相关系数，也称Pearson相关系数，是一个统计中常用的标准指标：

$$C(\mathbf{D},\mathbf{M})=\sum_i\frac{(d_i-\bar{d})(m_i-\bar{m})}{\sigma_\mathbf{D}\sigma_\mathbf{M}}, \tag{6.13}$$

其中，$\bar{d}=\sum d_i/N$，$\bar{m}=\sum m_i/N$，为各样本的平均值；$\sigma_\mathbf{D}$，$\sigma_\mathbf{M}$为相应的标准差。在

二级结构预测的研究中，这个指标在参考文献［382］中首次使用，因此也称做Matthews相关系数。相关系数总为取值在［-1, 1］上的实数，可以是非二值变量的形式。它所度量的是归一化参数（$d_i-\bar{d}$）/$\sigma_{\mathbf{D}}$和（$m_i-\bar{m}$）/$\sigma_{\mathbf{M}}$间的关联程度。取值-1代表完全负相关，取值+1代表完全正相关，取值为0表示预测结果完全是随机的。所以很容易将预测结果与随机的结果作比较。如果两个变量是彼此独立的，那么它们的相关系数也为0，但其逆命题却不一定为真。

如果用向量的形式表示，相关系数可以写做归一化向量内积的形式：

$$C(\mathbf{D},\mathbf{M})=\frac{(\mathbf{D}-\bar{d}\mathbf{1})(\mathbf{M}-\bar{m}\mathbf{1})}{\sqrt{(\mathbf{D}-\bar{d}\mathbf{1})^2}\sqrt{(\mathbf{M}-\bar{m}\mathbf{1})^2}}=\frac{\mathbf{DM}-N\bar{d}\bar{m}}{\sqrt{(\mathbf{D}^2-N\bar{d}^2)(\mathbf{M}^2-N\bar{m}^2)}} \tag{6.14}$$

其中**1**表示元素全为1的N维向量。这样，C（**D**, **M**）与L^2距离有关联，但由于本身可取负值，所以不属于距离度量。如果向量**D**和**M**经过归一化，则Q（**D**, **M**）=（**D**−**M**）^{2}=2−2**DM**=2−2C（**D**, **M**）。和前面提到的评价指标不同，相关系数更重视对全局性能的把握，而不仅是各位置点预测性能的叠加。

当**D**, **M**为0、1元素组成的向量时，有$\mathbf{D}^2=TP+FN$，$\mathbf{M}^2=TP+FP$，$\mathbf{DM}=TP$等成立。经过代数变换，可得

$$C(\mathbf{D},\mathbf{M})=\frac{TP-N\bar{d}\bar{m}}{N\sqrt{\bar{d}\bar{m}(1-\bar{d})(1-\bar{m})}} \tag{6.15}$$

对于属于螺旋类别的残基，有$\bar{d}$ =（$TP+FN$）/N，$\bar{m}$ =（$TP+FP$）/N，因此，

$$\begin{aligned}C(\mathbf{D},\mathbf{M})&=\frac{N\times TP-(TP+FN)(TP+FP)}{\sqrt{(TP+FN)(TP+FP)(TN+FP)(TN+FN)}}\\&=\frac{TP\times TN-FP\times FN}{\sqrt{(TP+FN)(TP+FP)(TN+FP)(TN+FN)}}\end{aligned} \tag{6.16}$$

可以看出，相关系数使用了TP、TN、FP和FN这四个值的信息，与百分比之类的指标比较，预测性能评价的可靠性更高。但在有些例子中，相关系数尚不能很好地评价系统的性能。例如，若预测算法的假阳性率极低或为0，则相关系数值相对较高，但同时被正确预测为正样本的数目也会很少。C值对于FP和FN是对称的，这一结论在后面的章节中会很有用。

相关系数的一个很有趣的特性是：可以通过简单的近似统计来检验相关系数值与0值的差异，即在$\bar{m}$相同的情况下，与随机猜测相比，预测结果和数据之间

的相关性是否显著提高。将χ^2检验应用到包含有TP、TN、FP和FN这四个值的2×2的矩阵上，可以很容易得到检验统计量为$\chi^2=N\times C^2(\mathbf{D},\mathbf{M})$。

6.7.6　近似相关系数

博塞特和吉哥[110]定义了“近似相关系数”指标。当$TP+FN$、$TP+FP$、$TN+FP$或$TN+FN$任意一者为0时（例如无任何阳性预测结果的情况），该指标用以弥补Matthews相关系数没有定义的不足。作为替代，当以上各相加项皆不为0时，使用平均条件概率（ACP），定义为：

$$ACP=\frac{1}{4}\left[\frac{TP}{TP+FN}+\frac{TP}{TP+FP}+\frac{TN}{TN+FP}+\frac{TN}{TN+FN}\right] \quad (6.17)$$

否则仅计算有意义的条件概率项的平均值即可。近似相关系数AC由ACP简单变换得到：

$$AC=2\times(ACP-0.5) \quad (6.18)$$

与C值相同，AC取值1、0、-1分别对应全正确、随机和全错误预测结果。博塞特和吉哥研究发现，其计算值与真实的相关系数值很接近。

实际上，上面所说的奇异情况并不存在，因为当任意一个相加项趋于0时，C值也趋向于0。而且从直觉上讲，仅包含一种类别的预测毫无疑义，该预测不能传达数据的任何信息。相反地，由于ACP的表达式将无定义的概率项删除，使得AC方法不恰当地引入了不连续性。因此，于无意义的预测结果相对应，以上方法无法保证指标值为0。而且由于AC没有简单的几何表达式，所以这种近似度量的意义不大，不鼓励使用该度量来衡量预测性能。

6.7.7　相对熵

相对熵，也称交叉熵或KL函数（Kullback-Leibler），是基于两个概率向量$\mathbf{X}=(x_1,\cdots,x_M)$和$\mathbf{Y}=(y_1,\cdots,y_M)$的对比计算得到的，向量$\mathbf{X}$、$\mathbf{Y}$满足$x_i$，$y_i\geqslant 0$，$\sum x_i=\sum y_i=1$。相对熵的具体定义如下：

$$H(\mathbf{X},\mathbf{Y})=\sum_{i=1}^{M}x_i\log\frac{x_i}{y_i}=-H(\mathbf{X})-\sum_i x_i\log y_i \quad (6.19)$$

其中$H(\mathbf{X})=-\sum x_i\log x_i$为一般的熵值定义，具体可参阅有关信息论的参考文献[342,341]。对于它的两个自变量，$H(\mathbf{X},\mathbf{Y})$总是非负凸的，而且当且仅当

X=**Y**时，值为0。严格来说，因为该函数不是对称的，所以不算距离度量。虽然很容易将它对称化，从而构建一个标准的距离度量，但这并不是必需的，上面的公式对于问题的研究已经足够了。如果**Y**=**X**+ε近似于**X**，通过简单的泰勒展开，可得

$$H(\mathbf{X},\mathbf{X}+\varepsilon)=-\sum_i x_i\left[\log\left(1+\frac{\varepsilon_i}{x_i}\right)\right]\approx\sum_i\frac{\varepsilon_i^2}{x_i} \tag{6.20}$$

特别地，若**X**遵循均匀分布，则相对熵近似为LMS误差形式。
回到二级结构预测问题，可以利用下式估计预测**M**的性能：

$$H(\mathbf{D},\mathbf{M})=\sum_{i=1}^{N}\left[d_i\log\frac{d_i}{m_i}+(1-d_i)\log\frac{(1-d_i)}{(1-m_i)}\right] \tag{6.21}$$

这对应每个位置i的相对熵的和。对于非二值数据（如结合亲和力），或只有**D**取二值时，以上公式同样能很好地评价系统性能。而当**M**取二值时，相对熵的FP+FN部分的计算值趋于无穷大，此时H（**D**, **M**）≈（FP+FN）→∞，则该计算公式不再适用。

6.7.8 互信息

考察概率向量分别为**X**=（x_1, ⋯, x_M）和**Y**=（y_1, ⋯, y_K）的两个随机变量X, Y。令Z=（X, Y）为笛卡尔积空间上的联合随机变量，对应的概率向量为**Z**。X, Y之间的互信息I（X, Y）或I（**X**, **Y**）定义为Z和积XY之间的相对熵：

$$I(X,Y)=H(Z,XY) \tag{6.22}$$

可知上式的值恒为正。利用贝叶斯统计的概念很容易理解互信息的含义：它利用先验和后验分布值的不同，表示当一个随机变量确定时，另一个随机变量不确定度的减少量。X的不确定度是由它的先验分布的熵$H(X)=\sum x_i\log x_i$决定的。一但确定Y取值为y，则X的不确定度将由它的后验分布的熵$H(X|Y=y)=\sum_x P(X=x|Y=y)\log P(X=x|Y=y)$决定。互信息是依赖于观测值$y$的随机变量。对所有可能的$y$的取值做平均，便得到条件熵

$$H(X|Y)=\sum_y P(y)H(X|Y=y) \tag{6.23}$$

因此，熵与条件熵的差值刻画了由确定的Y带给X的平均信息量。容易验证

$$I(X,Y)=H(X)-H(X|Y)=H(Y)-H(Y|X)=H(X)+H(Y)-H(Z)=I(Y,X) \quad (6.24)$$

或利用对应的概率分布

$$I(\mathbf{X},\mathbf{Y})=H(\mathbf{X})-H(\mathbf{X}|\mathbf{Y})=H(\mathbf{Y})-H(\mathbf{Y}|\mathbf{X})=H(\mathbf{X})+H(\mathbf{Y})-H(\mathbf{Z})=I(\mathbf{Y},\mathbf{X}) \quad (6.25)$$

再回到二级结构预测的问题，当**D**, **M**都为二值时，互信息度量为

$$\begin{aligned}I(\mathbf{D},\mathbf{M})=&-H\left(\frac{TP}{N},\frac{TN}{N},\frac{FP}{N},\frac{FN}{N}\right)\\&-\frac{TP}{N}\log\left[\frac{TP+FP}{N}\frac{TP+FN}{N}\right]-\frac{FN}{N}\log\left[\frac{TP+FN}{N}\frac{TN+FN}{N}\right]\\&-\frac{FP}{N}\log\left[\frac{TP+FP}{N}\frac{TN+FP}{N}\right]-\frac{TN}{N}\log\left[\frac{TN+FN}{N}\frac{TN+FP}{N}\right]\end{aligned} \quad (6.26)$$

或

$$\begin{aligned}I(\mathbf{D},\mathbf{M})=&-H\left(\frac{TP}{N},\frac{TN}{N},\frac{FP}{N},\frac{FN}{N}\right)\\&-\frac{TP}{N}\log\left[\bar{d}\bar{m}\right]-\frac{FN}{N}\log\left[\bar{d}(1-\bar{m})\right]\\&-\frac{FP}{N}\log\left[(1-\bar{d})\bar{m}\right]-\frac{TN}{N}\log\left[(1-\bar{d})(1-\bar{m})\right]\end{aligned} \quad (6.27)$$

（见参考文献［549］），其中 $\bar{d}=(TP+FN)/N$和 $\bar{m}=(TP+FP)/N$（同前），并且

$$H\left(\frac{TP}{N},\frac{TN}{N},\frac{FP}{N},\frac{FN}{N}\right)=-\frac{TP}{N}\log\frac{TP}{N}-\frac{TN}{N}\log\frac{TN}{N}-\frac{FP}{N}\log\frac{FP}{N}-\frac{FN}{N}\log\frac{FN}{N} \quad (6.28)$$

为一般意义上的熵。和相关函数一样，互信息更注重全局性能的把握，而不单单注重局部性能的叠加。很明显，互信息总是满足$0\leqslant I(\mathbf{D},\mathbf{M})\leqslant H(\mathbf{D})$。因此，在预测性能评价中，同样使用归一化的互信息[452,454]系数形式：

$$IC(\mathbf{D},\mathbf{M})=\frac{I(\mathbf{D},\mathbf{M})}{H(\mathbf{D})} \quad (6.29)$$

其中

$$H(\mathbf{D})=-\frac{TP+FN}{N}\log\left[\frac{TP+FN}{N}\right]-\frac{TN+FP}{N}\log\left[\frac{TN+FP}{N}\right] \quad (6.30)$$

或更明确简洁地表示为：$H(\mathbf{D}) = -\overline{m}\log\overline{m} - (1-\overline{m})\log(1-\overline{m})$。归一化的互信息满足$0 \leqslant IC(\mathbf{D}, \mathbf{M}) \leqslant 1$。当$IC(\mathbf{D}, \mathbf{M})=0$时，有$I(\mathbf{D}, \mathbf{M})=0$，对应随机预测的情况（**D**, **M**独立）；$IC(\mathbf{D}, \mathbf{M})=1$时，有$I(\mathbf{D}, \mathbf{M})=H(\mathbf{D})=H(\mathbf{M})$，对应预测结果完全正确的情况。和相关系数一样，互信息更注重全局性能的把握，而不单单注重局部性能的叠加。互信息是关于FP和FN对称的，但由于分母的原因，互信息系数不是对称的。

6.7.9 敏感度和特异度

对于预测算法的输出为连续情况的两类别预测问题，TP、TN、FP和FN的取值将依赖于如何选择阈值。一般来说，会考虑算法的假阳性率和假阴性率，从中取个折中。

在ROC（receiver operating characteristic，用户操作特性）曲线图中，阈值的选择可以通过考察截断阈值在一定范围变化时，随之相应变化的“命中率”（hit rate）和“假阳性率”的变化曲线加以确定。一般来说，“命中率”［敏感度，$TP/(TP+FN)$］与“假阳性率”［也称误极率（false alarm rate），$FP/(FP+TN)$］的增长趋势一致（见图8-10）。同样地，可以在一张相似的图或两张独立的图中显示敏感度［$TP/(TP+FN)$］和特异度［$TP/(TP+FP)$］随截断阈值取值变化时的关系曲线。

敏感度是正确预测正样本的概率，特异度则是正样本被正确预测的概率。在医学统计中，“特异度”有时包含另外的含义，即指负样本被正确预测的可能性：$TN/(FP+FN)$，等于1与假阳性率的差值。这里我们倾向于使用负样本的敏感度。

如果敏感度表示为$x=TP/(TP+FN)$，特异度表示为$y=TP/(TP+FP)$，则

$$TP+FP=\frac{TP}{y} \qquad TP+FN=\frac{TP}{x}$$

$$TN+FP=N-(TP+FN)=\frac{Nx-TP}{x}$$

$$TN+FN=N-(TP+FP)=\frac{Ny-TP}{y} \tag{6.31}$$

只要x，y均不等于0，TP就不会等于0。本质上，以上公式是用（TP，x，y，N）这套参数替代（TP，TN，FP，FN）这套参数。将新的参数代入（6.16）中，经过运算，可以得到以敏感度和特异度表示的相关系数表达式：

$$C(\mathbf{D},\mathbf{M})=\frac{Nxy-TP}{\sqrt{(Nx-TP)(Ny-TP)}} \qquad (6.32)$$

注意：该表达式关于x，y（即敏感度和特异度）是完全对称的，或等价于该表达式关于FP和FN（即假阳性与假阴性的数目）是对称的。事实上，给定TP，FP和FN的改变等价于x和y的改变。同样，互信息表达式（6.27）和互信息系数表达式（6.29）也可以用（TP，x，y，N）这套参数表示。互信息表达式关于x和y，即FP和FN，也是完全对称的。（但对于互信息系数，这一结论不成立。）

6.7.10　总　结

总之，在等价与独立的假设下，如果**D**和**M**是二值的，则TP、TN、FP和FN中包含了所有的性能评价信息。任何一种单值表示的性能指标必然丢失一些信息。汉明距离和二次距离在二值的情况下完全等价。这些距离和百分比、L^p距离一样，仅由TP、TN、FP和FN这四个值中的两个得出。相关系数和互信息系数则是从全部四个值中得出的，因此能更好地评价性能。在连续问题中，推荐将相关系数和相对熵作为性能评价指标。

第7章　隐马氏模型（HMM）：理论

7.1　简　介

在20世纪90年代，在应用序列比对方法[11,418]中，研究人员发现在新预测到的蛋白序列中，只有大约1/3与其他已知序列有明确的相似性。[80,224,155]而不完整的新序列或序列片断与其他已知序列的相似性更低。随着各类基因组、cDNA和其他测序计划的开展，尤其测序过程中产生的表达序列标签（expressed sequence tag, EST）的积累，大规模的片断数据库变得越来越实用。[200]在1997年初，GenBank数据库中约有一半的数据由片断数据构成。这些数据包含了人类基因组表达序列的绝大部分。对这些片断进行识别和分类，并从中挖掘更多有用的信息，自然引起人们极大的兴趣。

利用多重序列比对提取蛋白质家族的保守模式序列，已经成为提高数据库检索敏感度和效率的一种有效手段。[23,52,250,334,41,38]不同于传统的序列比对，保守模式序列中包含了更多信息，例如在整个序列家族中或多或少保守的残基及其位置信息，残基插人和删除的概率等。所有有关序列共有的保守特征的描述方法，如序列谱（profile）[226]、可变模式（flexible pattern）[52]和嵌段（block）[250]，都可以视为隐马氏模型（HMM）的具体应用。

在过去几十年中，基于HMM的另一类概率图模型被用于各类时间序列的建模，尤其是用于语音识别的时间序列建模。[359,439]以上模型在诸如离子通道记录（ion channel recording）[48]和光字符识别（optical character recognition）[357]的其他许多领域也有应用。HMM也早已被应用于计算生物学领域，包括DNA的编码/非编

码区建模[130]，DNA中蛋白质结合位点[352]和蛋白质超家族[553,351]等的建模。然而直到90年代中期，HMM才与其他机器学习技术结合，被系统地用于建模、比对和分析整个蛋白质家族和DNA区域。

HMM与神经网络、随机文法以及贝叶斯网络密切相关，或在某种程度上成为它们的特例形式。在本章中，我们将介绍HMM，并讲述如何将其视为第3章的多骰子模型的推广。这里将采用与第4章类似的思路讲述HMM的有关理论，尤其是概念的发展和机器学习算法。本章随后的各节将逐步应用这些算法，来解释如何使用HMM分析生物序列。更多的具体应用可参考第8章内容，HMM与其他各类模型的关系则留待后续章节论述。

7.1.1 HMM的定义

1阶离散的HMM是一个关于时间序列的随机生成模型，由有限状态集合S、离散的字符集A、转移概率矩阵$T=(t_{ji})$和生成概率矩阵$E=(e_{iX})$共同定义。一个隐马氏系统随机地从一个状态变化为另一个状态，同时生成字符集（alphabet）中的一个字符。当系统处于状态i时，系统转移到状态j的概率为t_{ji}，同时生成字符X的概率为e_{iX}。由此HMM可以被设想为两个与状态相关的骰子：一个状态转移的骰子和一个生成字符的骰子。基本的1阶马尔可夫方程假设指出：生成和转移过程都只取决于当前状态，而与历史无关。因为只有系统生成的字符才能被观察到，系统在状态之间的随机游走（random walk）无法被观察到，由此冠之以“隐”马氏模型。这一隐藏的随机游走可被视为观察不到的隐藏的或潜在的随机变量。

与神经网络类似，与非零的t_{ji}连接相关联的有向图也称为HMM的构架（architecture）。一般假设存在“初始”和“终止”两个特殊的状态，尽管它们对于HMM的理论并非必要。在t=0时刻，系统处于初始状态。当然可以选择以状态空间上的一个概率分布作为初始状态。转移概率和生成概率都是模型的参数。另一种等价的理论是使生成过程基于状态转移而不是基于状态本身。连续字符空间上的HMM也存在，但由于我们关注的焦点在生物序列的离散特性，故本书对此不做更多讨论。

图7-1给出了一个很简单的HMM的例子。我们设想有两个“DNA骰子”。第一个骰子代表生成概率向量（e_{1A}=0.25，e_{1C}=0.25，e_{1G}=0.25，e_{1T}=0.25）；第二个骰子代表生成概率向量（e_{2A}=0.1，e_{2C}=0.1，e_{2G}=0.1，e_{2T}=0.7）。转移概率如图中给出。假设我们观察到一个序列ATCCTTTTTTTCA。我们可以立即提出至少三个问题：由这个特定的HMM生成这一序列的可能性有多大？（即可能性问题）对于由给定HMM所产生的这一特定序列，最可能的转移和生成概率序列是什

么？（即解码问题）最后，假设转移和生成概率参数未知，如何利用观察到的特定序列估计这些参数的值？（即学习问题）建议读者先就以上例子尝试回答这些问题。对于这三个问题的一般性的精确算法将在后面几节中给出。下面先介绍几个使用不同HMM构架的生物学应用例子。

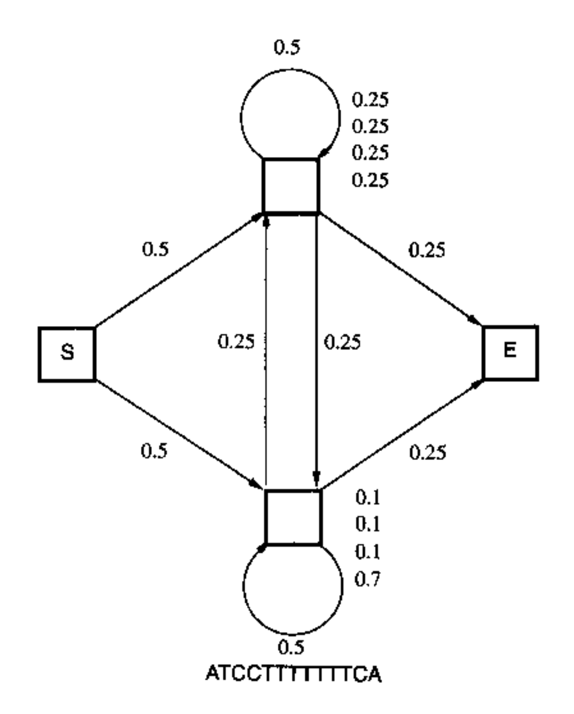


图7–1 HMM的一个简单例子（包括两个状态以及初始和终止状态）

7.1.2 生物序列的HMM

在生物序列分析中，HMM的字符集自然是构成蛋白质的20种氨基酸和构成DNA/RNA的4种核苷酸的字符。然而，也可以根据不同的问题使用其他多种字符集，例如包含64个符号的三联体符号表，表示二级结构的三元符号表（α、β、卷曲），以及由各种字符集的笛卡尔积生成的新的字符集（见表6–1）。如果需要，还可在任何字符集中添加1个空格符号。在本章和第8章中我们仅使用蛋白质和DNA的符号表。

在上述的简单HMM构架中，只包含两个隐状态，并在两个隐状态间建立了完全连接。在实际应用中，我们需要考虑更加复杂的HMM构架，它们包含更多的状态以及状态间的稀疏连接。设计和选择何种构架，在很大程度上是由所研究的问题决定的。生物序列分析中，在语音识别中常用的一种称为“从左到右”的构架，能够很好地抓住序列的线性特征。对于构架中的任一状态，一旦该状态转

移到其他状态，系统将不再返回该状态，这样的构架称为“从左到右”的。这里首先介绍在生物序列分析中广泛采用的最基本的“从左到右”构架，即标准线性构架（如图7-2所示）。

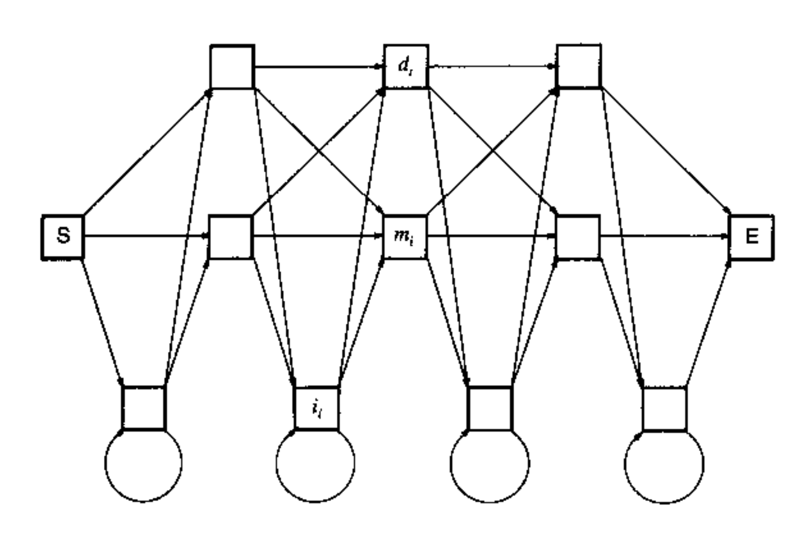


图7-2 标准HMM构架

S是初始状态，E是终止状态，d_i、m_i和i_i分别代表删除状态、主状态和插入状态。

首先从相关序列家族（如一个蛋白质家族）的建模开始。与HMM应用于语音识别类似，一个蛋白质家族可以被视为由一个HMM产生的同一个词的一系列不同发音。而这个标准HMM构架则可视为第3章介绍的多骰子模型的简单推广。多骰子模型实际上就是含有线性状态序列的简单HMM，其中每个骰子对应于一个状态。从一个状态到下一个状态的转移概率都设为1。每一个骰子的生成概率直接与序列家族中相应列的字符排列相关联。当然，字符的插入和删除是这个模型的主要问题：一般而言，同一家族中不同序列的长度N并不相同。即便在骰子的字符集中加入间隙符，我们仍然需要预先进行多重比对以决定每一个骰子的生成概率。标准HMM构架很简单，但它从根本上扩展了单骰子模型：在所有可能的位置上增加了对应于插入和删除操作的新状态。

在标准HMM构架中，除初始和终止状态外还有三种状态：主状态、插入状态和删除状态，表示为$S=\{start,m_1,\cdots,m_N,i_1,\cdots,i_{N+1},d_1,\cdots,d_N,end\}$。删除状态又称为间隙或跳跃状态。$N$是模型的长度，取值一般为家族中序列的平均长度。主状态和插入状态一般代表一个氨基酸字符，而删除状态则为“哑”状态。这相当于在字符集中加入一个空格符，并强制规定删除状态的生成符号仅为空格符。沿状态转移

的线性序列构成了模型的主干：起始→m_1→m_2…→m_N→终止。这些状态与多骰子模型的状态相对应。从每个主状态到插入和删除状态的转移过程，对应着相应序列字符的插入和删除过程。更准确地讲，删除过程应与主状态一一对应，而插入过程应与模型主干上的状态转移一一对应。插入状态的自环（self-loop）允许在同一位置插入多个字符。字符集大小为$|A|$的标准HMM构架，大致有$2N|A|$个生成概率参数和$9N$个转移概率参数，这里没有考虑细微的边界值误差［准确的数值为$(2N+1)|A|$个生成概率参数和$9N+3$个转移概率参数］。因此，当N足够大时，蛋白质模型的参数个数约为$49N$，DNA模型的参数个数约为$17N$。同样地，忽略边界的影响，有$2N$个生成概率归一化约束方程以及$3N$个转移概率归一化约束方程。

7.2　先验信息和初始化

在HMM的设计和参数选择中应用先验信息的方法有很多种。我们将在后面章节中列举不同的HMM构架。在选择了某一HMM构架之后，如果能获得相应的先验信息，就可以进一步限定部分参数的取值范围。这些先验信息包括高度保守模式和疏水区域信息。和神经网络模型中的权重共享一样，也可以将不同蛋白质的参数关联在一起。由于HMM的生成概率和转移概率都与多项式模型密切相关，所以Dirichlet分布自然成为HMM参数的先验分布。

7.2.1　转移概率矩阵的Dirichlet先验分布

在标准HMM构架中，Dirichlet分布$\mathcal{D}_{\alpha_i Q_i}(t_{ji})$很适合用于估计从状态$i$出发的转移概率向量$t_{ji}$。对于同一类的所有状态，我们可以用相同的Dirichlet分布，例如对于所有主状态（受边界影响的最后一个状态除外）。由此三个基本的先验分布——$\mathcal{D}_{\alpha_m Q_m}$、$\mathcal{D}_{\alpha_i Q_i}$和$\mathcal{D}_{\alpha_d Q_d}$——可以分别用于估计从主状态、插入状态和删除状态出发的转移概率向量。若需要，可以令$\alpha_m=\alpha_i=\alpha_d$，进一步化简超参数$\alpha$。需要注意的是，对于不同类型的状态Dirichlet分布的向量Q通常不一致，这是因为向主状态转移的概率期望比较大。

7.2.2　生成概率矩阵的Dirichlet先验分布

生成概率矩阵$\mathcal{D}_{\alpha_i Q_i}(e_{iX})$的情况是类似的。一个简单的方案是对于所有主状态和插入状态使用相同的Dirichlet先验分布。向量Q可以取值相同。另一种可行的方案是令Q等于训练集样本的平均组成频率。另外，某些文献还使用了混合Dirichlet分布。[334]

7.2.3 初始化

转移矩阵的初值一般取同一数值或随机给定。然而在标准HMM构架中，采用偏向于主状态的先验分布要比采用相同取值的初值效果更好。若在所有状态间进行转移的代价相同，则从主状态和插入状态出发的各种转移的代价也大致相同。结果插入状态最终可能非常频繁地被选中，这显然不是一个理想的方案。在参考文献［41］中，为了回避这个问题，引入了一个稍有不同的构架（如图7-3）。其中主状态的扇出度（fan-out）（3）比插入或删除状态的扇出度（4）稍低，而且比较所有用同一数值初始化状态转移概率的方案，达到主状态的代价也因此较小。生成概率矩阵可采用类似的方法初始化，如采取取值相同的初值、随机取值，甚至采用训练集样本组成的概率平均。如果采用Viterbi学习算法，任何显著偏离相同初值的初始化方案都可能引入不良的状态偏倚。

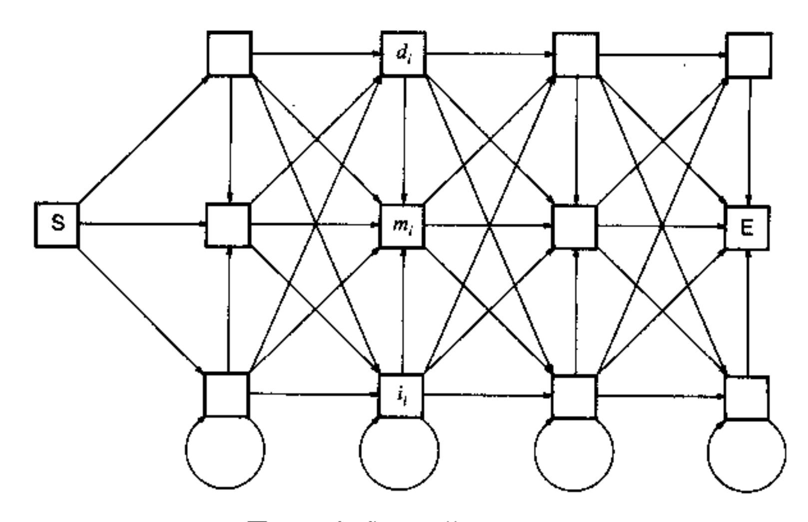


图7-3 标准HMM构架的变体

S代表初始状态，E代表终止状态，d_i、m_i和i_i分别表示删除状态、主状态和插入状态。

使用多重序列比对结果进行初始化

训练集的多重序列比对结果可以用来确定标准HMM构架的参数，至少能够在训练之前初始化这些参数，这一点很重要。若多重比对结果能提供更好的初值，我们当然希望初值能更靠近最优解，这样学习过程就会更快或者可以由此进一步得到更好的解。根据多重序列比对结果，如果比对中的某一列的间隙少于50%，可以将该列指定为一个主状态。若一列的间隙超过50%，可以相应地指定为一个

插人状态。删除状态则与位于间隙少于50%的比对列中的那些间隙相关联。主状态和插人状态的生成概率矩阵，可以根据相应列的频率计数值进行初始化，尽管这些初值仍然需要进行调整（根据Dirichlet分布或者它们的混合分布），以避免由零频率引人的生成矩阵的偏倚。类似的方法也可用于确定转移概率参数。

7.3 似然度及基本算法

在本节中，我们将研究一些HMM的基本算法，前面提出的三个问题中的前两个将在本节中会获得解答。尤其是如何计算似然度，以及与某一特定观测序列相对应的最可能的状态转移和生成序列。这些算法都是递归的，而且可视为某些形式的动态规划，或者与HMM相关的有向图上的传播算法。[439]将这些算法进行组合，构成了后续各节中所讲述的学习算法。引入删除状态将使一些公式变得稍微复杂些。

首先，假设问题是根据一个参数为w的HMM $M=M(w)$计算一个序列$O=\mathrm{X}^1\cdots\mathrm{X}^t\cdots\mathrm{X}^T$的可能发生概率$\mathbf{P}(O|w)$。定义$M$中的一条路径$\pi$为从初始状态到终止状态的一个彼此相连的状态序列，同时依次为该路径上的每一个生成状态（能够生成符号的状态）选择生成一个字符。若沿该路径所生成的字符序列与O相同，则

$$\mathbf{P}(O,\pi|w)=\prod_{start}^{end}t_{ji}\prod_{t=1}^{T}e_{i\mathrm{X}^t} \tag{7.1}$$

其中第一个乘积项处理路径π上的所有转移概率，而第二个乘积项则包括了路径π上生成状态i的生成概率。若沿路径生成的符号序列与序列O不相符，则显然有$\mathbf{P}(O,\pi|w)=0$。于是一个序列的似然度可以表示为

$$\mathbf{P}(O|w)=\sum_{\pi}\mathbf{P}(O,\pi|w) \tag{7.2}$$

然而，这个表达式并不能产生出一种计算似然度或其衍生结果的有效算法，因为在这一构架中，路径的数量是以指数增长的。幸运的是，有一种更为有效的计算似然度的算法，即“前向算法”。本节中的其他所有算法都与之类似，可视为一种沿构架进行的、遵循递归传播机制的计算方法。这种算法有效避免了逐一遍历所有隐藏的可能路径。

7.3.1 前向算法

我们定义

$$\alpha_i(t)=\mathbf{P}(S^t=i,\mathrm{X}^1\cdots\mathrm{X}^t|w) \tag{7.3}$$

为系统在时间t处于状态i的概率，假设观察到模型$M(w)$生成字符序列为$X^1 \cdots X^t$。可以将α初始化：

$$\alpha_{start}(0)=1 \tag{7.4}$$

若不存在初始状态，就要为所有状态给定一个初始概率，目标是计算$\mathbf{P}(O|w)=\alpha_{end}(T)$。其中$\alpha_i(t)$可以通过简单的传播过程递归地计算：

$$\alpha_i(t+1)=\sum_{j\in S}\alpha_j(t)t_{ij}e_{iX^{t+1}}=\sum_{j\in N^-(i)}\alpha_j(t)t_{ij}e_{iX^{t+1}} \tag{7.5}$$

以上邻接表示（neighborhood notation）又一次显示出一般性稀疏矩阵连接的好处。这个公式适用于所有生成状态，但对于删除状态，它需要稍做修改：

$$\alpha_i(t+1)=\sum_{j\in N^-(i)}\alpha_j(t+1)t_{ij} \tag{7.6}$$

初看起来，（7.5）和（7.6）并没有定义出一个正确的传播机制，因为在（7.6）中时间t+1同时出现在等式的两边。然而很容易看出：通过迭代计算，（7.5）和（7.6）一定会收敛到一组稳定的值$\alpha_i(t+1)$。对于标准HMM构架，由于不存在只经过删除状态的有向循环回路，这一结论是显而易见的。这种情况下，（7.6）最多只需被迭代N次。而即便在构架中存在经过删除状态的循环回路，（7.6）仍然是普遍收敛的，因为沿一个哑环路的概率传播，将形成一个比率为环路上转移概率乘积的几何级数，而这个比率值一般小于1（更多的细节请参阅附录D）。

若HMM中的一条从状态j到状态i的有向路径只包含对应于删除状态的内部节点，则该路径被称为“哑路径”。这样路径的概率为其所包含的转移概率的乘积。我们将从j到i的哑转移记为t_{ij}^D，因此t_{ij}^D是从j到i的所有哑路径的概率和。在标准HMM构架中，由于从j到i最多只能有一条哑路径，t_{ij}^D很容易计算。用这种表示法，前向传播可表示为首先由（7.5）计算所有生成状态的$\alpha_i(t+1)$。因此，删除状态的前向变量可由以下公式计算：

$$\alpha_i(t+1)=\sum_{j\in E}\alpha_j(t+1)\,t_{ij}^D \tag{7.7}$$

其中E为所有生成状态的集合。值得注意的是，（7.5）和（7.6）所定义的传播过程可以视为一个T层的线性神经元网络。网络的每一层对应一个时刻t，而每层包含M个单元，每个单元对应一个HMM状态。所有单元都是线性的。在第t+1层中，对应于生成状态i的单元有一个斜率为$e_{iX^{t+1}}$的线性转移函数。由此看来，HMM中

的似然度计算等价于一个大约包含N层、每层包含$|S|$个单元的线性网络中的前向传播计算过程。删除状态的出现在同一层中增加了一些连接。对标准HMM构架而言，尽管引入了这些层内连接，神经网络仍然是前馈型的，因此（7.6）在传播中仍是收敛的。由于算法的主要计算内容在于更新每层M个单元的T层网络，故前向算法的计算规模约为$O(MT)$。在标准HMM构架中，M和T的数量级都与N相同（$M\approx 3N$），因此前向传播算法的计算规模约为$O(N^2)$。

最后还需看到，应用前向变量的HMM可被视为一个动态混合模型。这是因为生成字符X^t的概率可被分解为$\sum_i \alpha_i(t)e_{iX^t}$。

7.3.2 后向算法

与神经网络类似，在学习算法中我们需要一个后向传播概率。后向算法与前向算法相反。我们定义后向变量为系统在时间t处于状态i时：观察到从X^{t+1}到结束的部分序列的概率

$$\beta_i(t)=\mathbf{P}(X^{t+1}\cdots X^T \mid S^t=i, w) \tag{7.8}$$

显然

$$\beta_{end}(T)=1 \tag{7.9}$$

递归地计算生成状态β的传播方程为

$$\beta_i(t)=\sum_{j\in N^+(i)} \beta_j(t+1)t_{ji}e_{jX^{t+1}} \tag{7.10}$$

对于删除状态，

$$\beta_i(t)=\sum_{j\in N^+(i)} \beta_j(t)t_{ji} \tag{7.11}$$

经过对生成状态变形，上述方程变为

$$\beta_i(t)=\sum_{j\in E} \beta_j(t)t_{ji}^D \tag{7.12}$$

前面对前向算法的说明都可以应用于后向算法。尤其对于标准HMM构架，后向算法的计算复杂度约为$O(N^2)$。

应用前向和后向变量，对于给定的观察序列O和模型w，我们很容易计算系

统在时间t处于状态i的概率

$$\gamma_i(t)=\mathbf{P}(S^t=i|O,w)=\frac{\alpha_i(t)\beta_i(t)}{\mathbf{P}(O|w)}=\frac{\alpha_i(t)\beta_i(t)}{\sum_{j\in S}\alpha_j(t)\beta_j(t)} \quad (7.13)$$

或在时间t发生$i\rightarrow j$转移的概率$\gamma_{ji}(t)$

$$\mathbf{P}(S^{t+1}=j,S^t=i|O,w)=\begin{cases}\alpha_i(t)t_{ji}\,e_{iX^{t+1}}\beta_j(t+1)/\mathbf{P}(O|w) & 若j\in E\\ \alpha_i(t)t_{ji}\,\beta_j(t)/\mathbf{P}(O|w) & 若j\in D\end{cases} \quad (7.14)$$

其中D为删除状态集合。显然还可以得到

$$\gamma_i(t)=\mathbf{P}(S^t=i|O,w)=\sum_{j\in S}\gamma_{ji}(t) \quad (7.15)$$

通过求$\gamma_i(t)$的极大值，我们可以确定时刻t最可能的状态。但在解码问题中，我们关心的是最可能路径。最可能路径对于构架学习和序列到模型的匹配也很有用。Viterbi算法可以用来计算最可能路径。它是动态规划的另一个应用，其本质与序列比对算法相同。

7.3.3 Viterbi算法

对于Viterbi算法，需要定义变量

$$\delta_i(t)=\max_{\pi_i(t)}\mathbf{P}[\pi_i(t)|w] \quad (7.16)$$

其中$\pi_i(t)$为生成$X^1\cdots X^t$并结束于状态i的“前缀”（prefix）路径。因此，$\delta_i(t)$是与最可能路径相关的概率，该路径产生结束于状态i的序列O的前t个字符。这些变量可用类似于前向算法的传播机制进行更新计算，其中求和运算被求极大值运算

$$\delta_i(t+1)=[\max_j\delta_j(t)t_{ij}]e_{iX^{t+1}} \quad (7.17)$$

所替代用于生成状态，而

$$\delta_i(t+1)=[\max_j\delta_j(t+1)t_{ij}]^{⑤} \quad (7.18)$$

则用于删除状态。与前向算法相比，其收敛性更显而易见；删除状态的循环回路

⑤ 原文如此，似有错误，疑为$\delta_i(t+1)=[\max_j\delta_j(t+1)t_{ij}]e_{iX^{t+1}}$。 ——编者注

根本不可能进入最佳路径，因为它将降低总概率而没有生成任何字符。为了恢复最佳路径，我们需要在每个时刻都保存能够回溯前一时刻最佳状态的数据。Viterbi算法获得的路径将被用于后续章节中的构架学习和多重序列比对过程。

7.3.4　数学期望的计算

对于给定参数集w和给定序列O，$\mathbf{P}(\pi|O, w)$定义了隐变量的一个后验分布$Q(\pi)$，即路径的概率π。在第3章和第4章中，我们已经看到后验分布Q扮演了一个重要的角色。尤其在学习的过程中，需要根据Q来计算数学期望，例如：达到状态i的次数的期望值，字符X从状态i生成的次数的期望值，以及$i \to j$转移发生的次数的期望值等。由于HMM的乘积因子特性，计算Q很方便，而且相关的期望值可以通过前向—后向变量获得。令

- $n(i, \pi, O)$ 为给定π和O，达到状态i的次数；
- $n(i, X, \pi, O)$ 为给定π和O，从状态i生成字符X的次数；
- $n(j, i, \pi, O)$ 为给定π和O，$i \to j$转移发生的次数。

于是，相应的各项期望值可计算如下：

$$n_i = \sum_{\pi} n(i, \pi, O)\mathbf{P}(\pi|O, w) = \sum_{t=0}^{T} \gamma_i(t) \tag{7.19}$$

$$n_{iX} = \sum_{\pi} n(i, X, \pi, O)\mathbf{P}(\pi|O, w) = \sum_{t=0, X^t = X}^{T} \gamma_i(t) \tag{7.20}$$

对于转移概率，类似的公式为

$$n_{ji} = \sum_{\pi} n(j, i, \pi, O)\mathbf{P}(\pi|O, w) = \sum_{t=0}^{T} \gamma_{ji}(t) \tag{7.21}$$

至此，我们已经掌握了求解HMM学习问题的全部工具。

7.4　学习算法

HMM的学习算法有多种，包括Baum–Welch算法或EM（期望最大化）算法，以及不同形式的梯度下降和其他GEM（广义期望最大化）算法。[147,439,39] 当然，还可以应用模拟退火算法，尽管用它处理大模型不太现实。与通常一样，我们将集中讨论1阶贝叶斯推断：通过最大后验估计（MAP）寻求最优参数。我们首先

使用最大似然估计法估计生成概率参数，转移概率参数的计算与之类似。首先假设训练集只包含惟一序列O；对于每种学习算法，先推导其在线学习方式的最大似然估计公式；然后简要介绍如何将这些公式修改并扩展到多序列成批学习中，以及何时引入先验分布（应用MAP法）。当存在K个训练序列时，它们可被视为彼此独立的，训练集的总概率等于各序列概率的乘积。对于HMM，高阶贝叶斯推断目前还很少被用到，甚至使用率少于神经网络，因此这里仅做简要介绍。

仍假设概率$\mathbf{P}(O\mid w)=\sum_{\pi}\mathbf{P}(O,\ \pi\mid w)$。在最大似然估计中，希望使拉格朗日算子最优化：

$$\mathcal{L}=-\log\mathbf{P}(O\mid w)-\sum_{i\in E}\lambda_i\left(1-\sum_{\mathrm{X}}e_{i\mathrm{X}}\right)-\sum_{i\in S}\mu_i\left(1-\sum_{j}t_{ji}\right) \tag{7.22}$$

其中λ和μ为大于0的拉格朗日算子。由（7.1）可得

$$\frac{\partial\mathbf{P}(O,\pi|w)}{\partial e_{i\mathrm{X}}}=\frac{n(i,\mathrm{X},\pi,O)}{e_{i\mathrm{X}}}\mathbf{P}(O,\ \pi\mid w) \tag{7.23}$$

通过令拉格朗日算子的偏导数为0求极值，可以推得

$$\lambda_i e_{i\mathrm{X}}=\sum_{\pi}n(i,\ \mathrm{X},\ \pi,\ O)Q(\pi)=n_{i\mathrm{X}} \tag{7.24}$$

类似地，可得相应的转移概率参数。Q表示后验概率$\mathbf{P}(\pi|O,\ w)$。通过对整个字符集求和可得

$$\lambda_i=\sum_{\pi}\sum_{\mathrm{X}}n(i,\ \mathrm{X},\ \pi,\ O)Q(\pi)=\sum_{\pi}n(i,\ \pi,\ O)Q(\pi)=n_i \tag{7.25}$$

于是，在极值点有

$$e_{i\mathrm{X}}=\frac{\sum_{\pi}n(i,\mathrm{X},\pi,O)\,Q(\pi)}{\sum_{\pi}n(i,\pi,O)\,Q(\pi)}=\frac{\sum_{\pi}\mathbf{P}(\pi|O,w)\,n(i,\mathrm{X},\pi,O)}{\sum_{\pi}\mathbf{P}(\pi|O,w)\,n(i,\pi,O)} \tag{7.26}$$

以上最大似然方程不能直接求解，因为后验分布Q依赖于$e_{i\mathrm{X}}$的值。然而（7.26）给出了一种简单的迭代算法：首先通过$Q(\pi)=\mathbf{P}(\pi\mid O,\ w)$估计$Q$，然后应用（7.26）更新参数，这恰恰就是HMM的EM算法。

7.4.1 EM算法（Baum–Welch算法）

在第4章讨论EM算法时，我们定义了隐层上的能量函数$f(\pi)=-\log\mathbf{P}(O,\ \pi\mid w)$。

EM算法可以看做是对函数$\mathcal{F}(w, Q)=\mathbf{E}_Q(f)-\mathcal{H}(Q)$（温度为1时的自由能函数）的最小化双迭代过程，即先对Q再对w进行。第一步最小化迭代过程产生后验概率$Q(\pi)=\mathbf{P}(\pi|O, w)=\mathbf{P}(\pi, O|w)/\mathbf{P}(O|w)$，其计算方法是我们所熟知的。在第二步最小化迭代过程中，需要对于不同的w最小化$\mathcal{F}$，同时需要满足概率归一化的约束。由于熵这一项与w无关，最终需要在$Q(\pi)=\mathbf{P}(\pi|O, w)$不变的情况下，极小化拉格朗日算子

$$\mathcal{L}=\mathbf{E}_Q(f)-\sum_{i\in E}\lambda_i(1-\sum_{\mathrm{X}}e_{i\mathrm{X}})+\sum_{i\in S}\mu_i(1-\sum_j t_{ji}) \tag{7.27}$$

应用（7.23）可得

$$\lambda_i e_{i\mathrm{X}}=\sum_{\pi}n(i, \mathrm{X}, \pi, O)Q(\pi)=n_{i\mathrm{X}} \tag{7.28}$$

再对整个字符集求和可得

$$\lambda_i=\sum_{\pi}\sum_{\mathrm{X}}n(i, \mathrm{X}, \pi, O)Q(\pi)=\sum_{\pi}n(i, \pi, O)Q(\pi)=n_i \tag{7.29}$$

这些公式与（7.24）和（7.25）相同。可以进一步验证它们对应于某一极小值，于是EM的再估计公式为

$$e_{i\mathrm{X}}^{+}=\frac{\sum_{\pi}n(i,\mathrm{X},\pi,O)Q(\pi)}{\sum_{\pi}n(i,\pi,O)Q(\pi)}=\frac{\sum_{t=0,\mathrm{X}^t=\mathrm{X}}^{T}\gamma_i(t)}{\sum_{t=0}^{T}\gamma_i(t)}=\frac{n_{i\mathrm{X}}}{n_i} \tag{7.30}$$

对于转移概率参数，类似地我们能得到

$$t_{ji}^{+}=\frac{\sum_{\pi}n(j,i,\pi,O)Q(\pi)}{\sum_{\pi}n(i,\pi,O)Q(\pi)}=\frac{\sum_{t=0}^{T}\gamma_{ji}(t)}{\sum_{t=0}^{T}\gamma_i(t)}=\frac{n_{ji}}{n_i} \tag{7.31}$$

因此，EM算法是用前向和后向过程来实现的。实际上，HMM的EM算法有时又被称为前向—后向算法。$e_{i\mathrm{X}}^{+}$是在状态i观察到X的次数的期望除以系统到达状态i的次数的期望；而t_{ji}^{+}则是从状态i到j转移的次数的期望除以所有从状态i出发的转移的次数的期望。这些迭代公式与前面令概率的拉格朗日算子的导数为0所得到的（7.22）完全相同。这是HMM和乘积因子型分布的一个特性，并不是一个普遍的规律。

在考虑K个序列$O_1, \cdots, O_K$时，类似的公式可写为

$$e_{iX}^{+}=\frac{\sum_{j=1}^{K}\sum_{\pi}n\left(i, X, \pi, O_j\right)\mathbf{P}\left(\pi|O_j, w\right)}{\sum_{j=1}^{K}\sum_{\pi}n\left(i, \pi, O_j\right)\mathbf{P}\left(\pi|O_j, w\right)} \tag{7.32}$$

可以很方便地修改这些公式以便将EM算法应用于MAP估计。每个训练序列都需要一个前向和一个后向传播过程。因此，EM算法的计算规模约为$O(KN^2)$。

可成批计算的EM算法已被广泛应用于HMM学习。然而，我们需要意识到在线应用HMM仍然存在问题。这主要由于EM算法与梯度下降法不同，没有调节学习率的手段。受每个孤立的训练样本影响，EM算法会以较大的迭代步幅沿递减方向收敛于$\mathcal{E}$的不甚理想的局部极小值。在梯度下降法中，这种“地毯式跳跃”（carpet-jumping）效应可以通过采用较小的学习率得以避免。

7.4.2 梯度下降法

负对数似然度的梯度下降公式，可以利用HMM和神经网络的关联以及反向传播公式推导得出。在这里，我们将采用参数替换方法（reparameterization）直接推导该公式，该方法等价于使用归一化约束的拉格朗日算子的方法。我们用以下形式的归一化的指数函数对HMM进行参数替换：

$$e_{iX}=\frac{e^{w_{iX}}}{\sum_{Y}e^{w_{iY}}} \quad 和 \quad t_{ji}=\frac{e^{w_{ji}}}{\sum_{k}e^{w_{ki}}} \tag{7.33}$$

其中w_{iX}和w_{ji}是新的变量。这一参数替换有两个优点：（1）改变参数w，可自动保证转移和生成概率分布的归一化约束；（2）转移和生成概率永远不为0。通过简单的运算得出

$$\frac{\partial e_{iX}}{\partial w_{iX}}=e_{iX}\left(1-e_{iX}\right) \quad 和 \quad \frac{\partial e_{iX}}{\partial w_{iY}}=-e_{iX}e_{iY} \tag{7.34}$$

转移概率参数与此类似。利用链规则，

$$\frac{\partial \log \mathbf{P}\left(O|w\right)}{w_{iX}}=\sum_{Y}\frac{\partial \log \mathbf{P}\left(O|w\right)}{e_{iY}}\frac{\partial e_{iY}}{w_{iX}} \tag{7.35}$$

进而，利用（7.2）、（7.23）和从（7.33）到（7.35）的各式中得到负对数似然度的在线梯度下降公式

$$\Delta w_{iX}=\eta(n_{iX}-n_i e_{iX}) \quad 和 \quad \Delta w_{ji}=\eta(n_{ji}-n_i t_{ji}) \tag{7.36}$$

其中η为学习率。对于在线应用，n_{iX}和n_{ji}是从每个单一序列的前向—后向过程导出的次数的期望值。通过对所有训练序列求和，我们可以很容易地导出成批学习的梯度下降公式。对于MAP估计，需要在在线学习的梯度下降公式中添加对数先验概率对w的偏导数。例如，每个参数的高斯先验分布都需要在（7.36）中加入一个衰减权重项。

和EM算法一样，梯度下降公式要求进行一次前向传播和一次后向传播。因此每个训练循环需要$O(KN^2)$次计算。在实现中要特别小心，尽量减少归一化指数参数计算引入的开销。与EM算法不同，在线梯度下降法是一个平滑算法（smooth algorithm）。参考文献［39］中讨论了其他多种平滑算法。平滑算法的一个有用特性是很容易对错误样本进行校正。如果训练集中偶然包含有一个错误序列（例如该序列不属于被建模的序列家族），我们对该序列的梯度下降计算结果进行补偿，便可很容易地去除其对模型的负面影响。

7.4.3 Viterbi学习算法

EM算法和梯度下降法的迭代更新公式都基于计算所有隐含路径的数学期望。一般性的Viterbi学习算法的基本思想是只对少数可能路径进行计算，以代替对全部可能路径的计算；一般地，仅计算每个序列中最可能的一条路径。因此，对所有路径进行平均的生成次数$n(i, X, \pi, O)$被单一值$n(i, X, \pi^*(O))$——沿最可能的路径$\pi^*(O)$，状态i生成字符X的次数所代替。在标准HMM构架中，$n(i, X, \pi^*(O))$总为0或1，除非对于插入状态——由于反复插入同一字符，它偶尔会大于1。因此，简单的在线Viterbi EM算法没什么意义，因为模型参数在绝大多数情况下仅更新为0或1。在在线Viterbi梯度下降法中，每一步沿Viterbi路径上的任意状态i，模型参数将根据以下公式进行更新：

$$\Delta w_{iX}=\eta(E_{iX}-e_{iX}) \quad 和 \quad \Delta w_{ji}=\eta(T_{ji}-t_{ji}) \tag{7.37}$$

若状态i生成字符X（对应转移$i\to j$），则$E_{iX}=1$（对应$T_{ji}=1$），否则$E_{iX}=0$。因此，将根据由训练集获得的频率与模型的概率参数之间的差值，对参数进行更新。

在一些文献中，利用Viterbi学习算法快速近似相应的非Viterbi算法。实际上，Viterbi学习算法在速度方面的优势并不显著，其速度只是其他算法的2倍左右，因为Viterbi算法在计算$\pi^*(O)$时不再需要反向传播过程。至于近似程度，Viterbi算法比较粗糙，因为它的序列似然度一般不会在最佳路径附近形成尖锐的峰值。因此，无论是在学习过程中还是在最终的模型中观察到Viterbi算法和非Viterbi算法之间

存在显著差异并非偶然。根据经验，Viterbi算法在蛋白质家族建模中通常能得到好的结果，但在DNA基本组成建模（如外显子区或启动子区建模）中则不理想，在这些问题中使用非Viterbi算法的结果更好。部分原因可能在于最佳路径在蛋白质中扮演着特别的角色。

实际上，可以从另一个角度考察Viterbi算法的本质：优化一个不同的目标函数。我们可以定义一个新的概率测度P^V，进而在序列空间上定义一个新模型（隐Viterbi模型）：

$$P^V(O|w)=\frac{\mathbf{P}\left(\pi^*(O)|w\right)}{\sum_O \mathbf{P}\left(\pi^*(O)|w\right)} \tag{7.38}$$

Viterbi算法变为极小化以下公式：

$$\mathcal{E}=\sum_{k=1}^{K}-\log P^V\left(O_k|w\right) \tag{7.39}$$

需要注意，随参数w变化，最佳路径π^*会随之不连续地变化，从而导致$\mathcal{E}$也不连续。显然，在MAP估计中使用Viterbi算法，可以为（7.39）添加一个正则项。

7.4.4 HMM学习算法的其他问题

一般地，当我们考虑改进学习算法时会提出许多其他问题，例如：如何平衡训练集，[157,337]改变学习率，如何通过估计Hessian似然矩阵来利用2阶信息等。已有许多文献讨论这些问题，鉴于篇幅，本书对此不再详述。然而，对于标定、构架的选择和学习，以及歧义符号等具有重要实际意义的问题，我们仍在此给予简要讨论。

标　定

概率$\mathbf{P}(\pi|O, w)$是很多转移概率和生成概率的乘积。由于乘积的每一项都小于1，所以一般来说它的值很小。对于大部分模型，这一概率值将超出所有机器的精度范围，即便采用双精度浮点数也是如此。因此在实现学习算法时，尤其是前向和后向算法，我们将面临精度下溢问题。这些问题可以通过标定过程加以解决，即在传播过程中对前向和后向变量进行标定以避免精度下溢。标定过程的技术性比较强，我们将在附录D中详细介绍。在Viterbi学习中，采用对概率取对数的方法，可以很容易地解决精度问题。

模型构架的学习

一个很自然的问题是能否通过训练数据获得HMM的构架。目前已有一些通

过学习建立一般化HMM构架的算法，例如参考文献［504］中介绍的方法。研究人员甚至结合生物序列上下文先验知识，提出了专门的构架学习方法。[193]参考文献［504］中提出的方法的基本想法是从一个非常复杂的模型开始，基本上是每个字符一个状态，然后经过迭代合并状态。选择合并状态以及终止条件取决于后验概率的评价。而在参考文献［193］中，初始模型为一个小规模的完全连接的HMM。该算法在迭代过程中删除概率很低的转移，并且复制连接最多的状态，直到最佳路径的概率或后验概率值达到一个足够的水平。这两种算法都在状态数不超过50的小规模HMM模型上获得了理想的实验结果。这些方法可能对于某些问题有用，然而它们的速度都很慢，在目前的计算机上很难实际应用。在第8章中我们将看到，大多数大规模HMM都没有任何先验知识可以利用。拥有$|S|$个状态的模型的所有可能的构架数目当然是极大的。相比之下，有关构架学习的一个更具操作性的特例是通过学习确定标准HMM构架的长度N。

模型长度的自适应调节

到目前为止，在有关标准HMM构架的方法中，都将N设为建模序列的平均长度。实际上，采用这种简化方法效果很好。很自然，若训练之后发现这个长度N似乎不是最佳值，可以选取一个新的长度值重新开始训练。

在参考文献[334]中介绍了一种称为“外科”的算法（surgery algorithm），用于在训练过程中动态调整HMM的长度。这种算法的思想是根据构架的整体连接模式，在必要时添加或删除状态。如果在建模序列家族中有超过50%的序列用到某个插入状态，意味着该插入状态将在超过50%的相应的Viterbi路径中出现，于是会在相应的位置建立一个新的主状态，同时建立与新的主状态配合的插入和删除状态。新状态的生成概率和转移概率可初始化为相同的数值。类似地，如果某个删除状态在多于50%的序列中用到，相应的主状态以及与之配合的插入和删除状态可以被一并删除。剩下的构架的左侧未被改变，因此训练过程可继续进行。尽管这一方法未被证明总能收敛到一个稳定的长度，但在实际应用中它似乎总是收敛的。

构架的变体

正如前面已经提到的，还有许多与标准HMM构架相关的HMM构架经常被应用于分子生物学中。其中，多重HMM构架（如图8-5）用于分类，环状HMM构架（如图8-16）和轮状HMM构架（如图8-17）用于周期型模式的建模。标准HMM构架还可用于蛋白质二级结构的建模，[187]也可用于为带有相似折叠类型和功能的蛋白质建立二级结构的保守模式库。其他一些HMM构架已被用于原核

生物[336]和真核生物[107]的基因检测。第8章将给出一些特定的应用例子。

歧义符号

由于测序技术尚不完美，偶尔会出现一些歧义符号。例如，在DNA序列中X代表A、C、G、T四种可能，在蛋白质序列中B代表天冬酰氨酸或天冬氨酸。利用HMM，这类符号很容易通过多种方法处理。在数据库搜索中，使用"可疑者获益"（benefit of the doubt）策略是谨慎的做法。应用这种策略，在计算序列似然度和Viterbi路径时，一个歧义符号被其最可能的符号候选代替。另外，应该注意那些歧义符号比例高得不正常的序列，因为它们很可能导致假阳性。

7.5 HMM的应用：一般性的问题

无论采用哪种设计和训练方法，一旦由序列家族成功地得到HMM，便可将其用于一系列不同的任务，包括：

1. 多重序列比对；
2. 序列和片段的数据挖掘和分类；
3. 结构分析和模式识别。

所有这些任务都是通过对给定序列计算与模型和最可能路径相关的概率，以及分析模型结构本身完成的。在大多数情况下，使用HMM技术能够很好地完成这些任务，例如多重序列比对的结果可以和专家人工完成的结果相媲美。有关HMM应用于蛋白质和DNA分析领域中的详细例子将在第8章中讨论。可以将HMM的模型库组织成层次或模块结构，以便产生逐步精细化的序列空间区域的概率模型。理论上，HMM可以用来生成与目标家族有很高相似性的"全合成序列"（de novo sequence），尽管这一特性尚未得到实际应用。

7.5.1 多重序列比对

计算一个序列的Viterbi路径也称为"序列到模型的匹配"。通过一种高效的方法两两比对多条Viterbi路径，可以生或多重序列比对数据。[334,41] 由于训练一个模型往往需要较长的时间，可以脱机进行。一旦完成训练过程，进行K个序列的多重比对仅需计算K条Viterbi路径，因此计算规模大约只有$O(KN^2)$。这一规模对K是线性的，显著优于多维线性规划比对算法的计算规模$O(N^K)$，它对K是指数的。而且从某种意义上讲，由HMM生成的多重序列比对数据比传统比对方法产生的结果更丰富。实际上，考虑用传统方法对两个序列进行比对，假

设相对于第一个序列，第二个序列在给定位置有一个间隙。这个间隙可以来自第二个序列上的一个删除过程，也可以来自第一个序列上的一个插入过程，它们分别对应HMM中两组不同的Viterbi路径，而传统的比对方法无法区分它们。

可以从另一个角度考察这一问题：设想传统的多重序列比对可以通过训练一个类似于标准HMM构架的HMM构架产生，但其中模型的长度固定设为最长匹配序列的长度，而且去掉所有的插入状态，仅留下主状态和删除状态。因此，所有Viterbi路径只由主状态的生成过程和对应于主状态的间隙构成。但在任何情况下，由同时包含插入和删除状态的HMM生成的多重序列比对结果显然要更加丰富。而且事实上多重序列比对结果应该用三维空间表示，而不是使用传统多重序列比对中的二维表示（第三个维度是为生成过程专门设置的）。由于绘图上的困难和人们的习惯，像传统的比对一样，基于HMM的比对仍然被绘制在二维图上，而小写字母经常被留做表示由HMM插入状态生成的字符。

HMM的插入和删除状态与序列的形式操作相对应。一个重要的问题是，它们是否以及如何与进化中的事件相关。这个问题当然关系到系统进化树的构建及其与HMM、多重序列比对的关系。标准HMM构架本身并没有为进化过程提供一个好的概率模型，因为它缺乏进化过程所需的树状结构，对替换过程也缺乏一种清晰的表示（除插入和删除外）。第10章将论及进化的概率模型。

需要提醒的是：上述的HMM多重序列比对仅仅基于单个的HMM，因而只是完整的贝叶斯处理过程的第一步。即便对于一个很简单的问题，如两个序列中的两个氨基酸能否相互匹配，一个完整的贝叶斯处理过程也要求根据某一后验分布的概率值在所有HMM上进行积分以给出最终答案。就目前所知，尚未在生物序列的HMM上计算过这一积分值（见参考文献［583］）。能否从这个对计算要求极高的实践扩展中得到更多的收获，对此我们没有把握。

最后，HMM还可以和替换矩阵一起使用。[27] HMM的生成概率分布可以用于计算替换矩阵，而替换矩阵又会在HMM训练中和训练之后影响HMM。对于规模大的训练集，我们可以认为大部分替换信息已在数据中出现，附加这类外部信息将不产生显著的好处。

7.5.2 数据挖掘和分类

给定一个训练好的模型，可以计算任意给定序列（及其相关的Viterbi路径）的概率。这些概率分值可用于判别和数据库检索，[334,38] 从而将与待训练序列家族相关的序列从数据库中抽取出来。这一方法可用于全序列和序列片断。[42] 有一个重要的问题将留待第8章进一步考察，即这个分值必须被标定为序列长度的函

数。

HMM还应用于分类问题，例如对多个蛋白质家族或一个蛋白质家族的多个亚家族进行分类。如果能够得到特定类别的训练集，这个问题可以通过为每个类别训练一个模型来解决。我们曾经用这种方法建立了两个HMM模型，它们可以非常可靠地判别酪氨酸和丝氨酸/苏氨酸激酶亚家族。否则可以将无监督聚类算法与HMM相结合，对数据集进行分类，如珠蛋白亚家族的分类（见参考文献［344］及第8章）。一般认为蛋白质亚家族的总数相对较少，大约为数千个左右。[127,93] 无论从算法还是计算能力的角度看，为每个蛋白质家族训练一个HMM，进而构成覆盖所有蛋白质家族的一个分类系统已经成为可行。这种全局分类系统正在开发中，并将成为其他许多研究问题的有用辅助工具，例如基因检测、蛋白质分类以及结构/功能预测。[497]

7.5.3 结构分析和模式识别

通过考察训练得到的HMM的结构，可以获得更多的信息和发现新的模式。我们可以用和研究神经网络连接相同的方法，研究一个HMM的参数。较高的生成或转移概率经常与结构/功能上很重要的一些保守区域或模式有关。检测这些模式的一个便捷办法是，沿模型的主干方向计算生成概率分布的信息熵。其他任何与位置有关的函数，例如疏水性和可弯曲性，也可以利用HMM概率进行平均并绘制成图。家族的特征模式，例如蛋白质二级结构的特征（α螺旋的疏水性）和DNA中的高可弯曲性区域，更容易在这样的图中被识别出来。这是由于个别序列的变化在计算数学期望时已被平滑掉了。其他一些模式，如周期性，则可以通过分析模型的结构将其揭示出来。标准HMM构架在模式检测方面的先天不足，将引导我们设计更有针对性的构架，例如轮状构架和环状构架，以加强周期性的信号。另外，使用HMM能够在未经比对的原始数据中发现微弱的模式，我们将在第8章中给出几个对比例子加以说明。

第8章　隐马氏模型（HMM）：应用

8.1　在蛋白质方面的应用

HMM已被成功地应用于许多蛋白质家族，例如：珠蛋白、免疫球蛋白、激酶以及G蛋白偶联性受体（G-protein-coupled receptor）（见参考文献［334,41,38］）。HMM还被应用于蛋白质二级结构要素的建模，如α螺旋和蛋白质超级家族二级结构的保守模式。[187]事实上，蛋白质家族数据库（Pfam）[497]和蛋白质家族的二级结构数据库（FORESST）[187]早在1997年底就已建成。来自HMM的多重序列比对数据也已出现在一些文献中。对于大量的HMM比对试验及其结果，我们在此不再一一赘述。需要指出的是，研究人员发现在大多数情况下，基于HMM的比对计算结果很好，其精度能够达到结构或系统进化信息（phylogenetic information）的人工多重序列比对的误差限度。在本章前半部分中，我们将依照参考文献［38］和［42］的思路，重点介绍HMM在一个特定蛋白质家族——G蛋白偶联性受体（GCR或GPCR）中的应用。更多的细节请参阅相关文献。

8.1.1　G蛋白偶联性受体

G蛋白偶联性受体是一个多样性的跨膜蛋白质家族，它们能够转导由激素、神经递质（neurotransmitter）、气味（odorant）和光等承载的多种细胞外信号（近期的有关综述见参考文献［436,325,508,227,552］）。尽管我们对这个家族所有成员的转导功能的具体生理机制尚未完全了解，但我们知道在大多数情况下，这类受体能激活一种鸟嘌呤核苷酸结合（G）蛋白。[402]我们相信该家族中的所

有受体都有类似的结构，它的特征是7个穿越疏水性膜间距的α螺旋。这7个跨膜区通过3个细胞外的环和3个细胞内的环相连接。受体的N端（amino termini）在细胞外，并经常被糖基化（glycosylated），而C端（carboxyl termini）在细胞质中，通常被磷酸化（phosphorylated）。这些螺旋的精确三维绞合方式和一般性三级结构，还有待进一步了解。[47,420]

我们经常根据递质的类型把这个家族分为一些亚族，例如毒蕈碱受体（muscarinic receptor）、儿茶酚胺受体（catecholamine receptor）和气味受体（odorant receptor）等等。从方法论角度看，GPCR家族颇具研究挑战性，其成员的长度差异很大，平均长度也相当大：已知的GPCR长度从200到1 200个氨基酸不等。该家族具有高度多样性，某些成员间共同的残基不足20%。

8.1.2 结构特性

在参考文献［38］中，从PROSITE数据库[23]中抽取的142个GPCR序列用于训练一个N=430（训练集中序列的平均长度）的HMM构架，具体方法是在整个训练集上应用在线Viterbi学习算法并经过12个迭代周期。

作为结构特性的一个例子，图8-1给出了相应模型主状态上生成概率分布的信息熵。信息熵的振幅谱中所包含的7个主要的波动与7个跨膜区域直接相关。于是我们得到了第一个近似推断：那些疏水区域比较稳定，因而与信息熵较低的区域相关。HMM在没有任何关于α螺旋或疏水性的先验知识的情况下，成功地发现了这一结构特性。

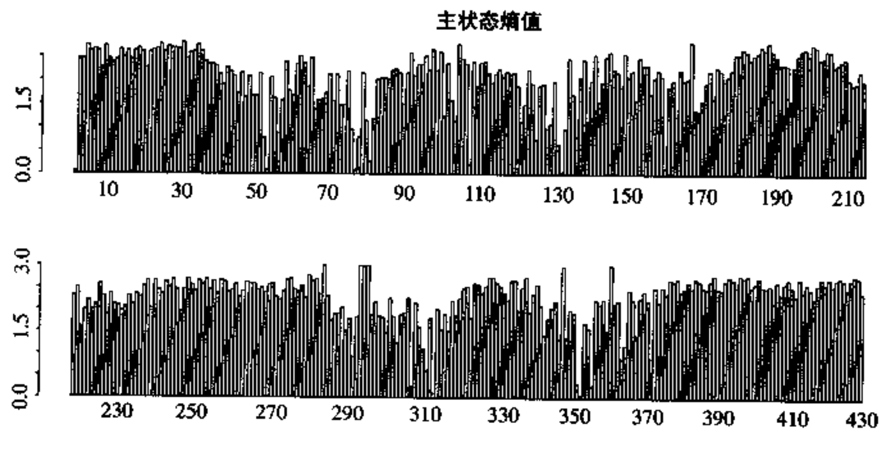


图8-1 经12次迭代获得的与HMM主状态相关的生成概率分布的熵谱

8.1.3 原始分值的统计

为了测试模型的鉴别能力，我们用和GPCR相同的平均成分组合生成了一个1 600个随机序列的测试集：长度为300、350、400、450、500、550、600、650、700、750、800、1 000的14组序列，每组100个序列；长度为1 500和2 000的2组序列，每组200个序列。对于任意序列，无论随机与否，其原始分值都通过模型计算获得。某一序列O的原始分值是对应于相关Viterbi路径概率的负对数。所有随机序列的原始分值、GPCR训练集中序列的分值以及SWISS-PROT数据库中全部序列的分值都被描绘在图8-2中。

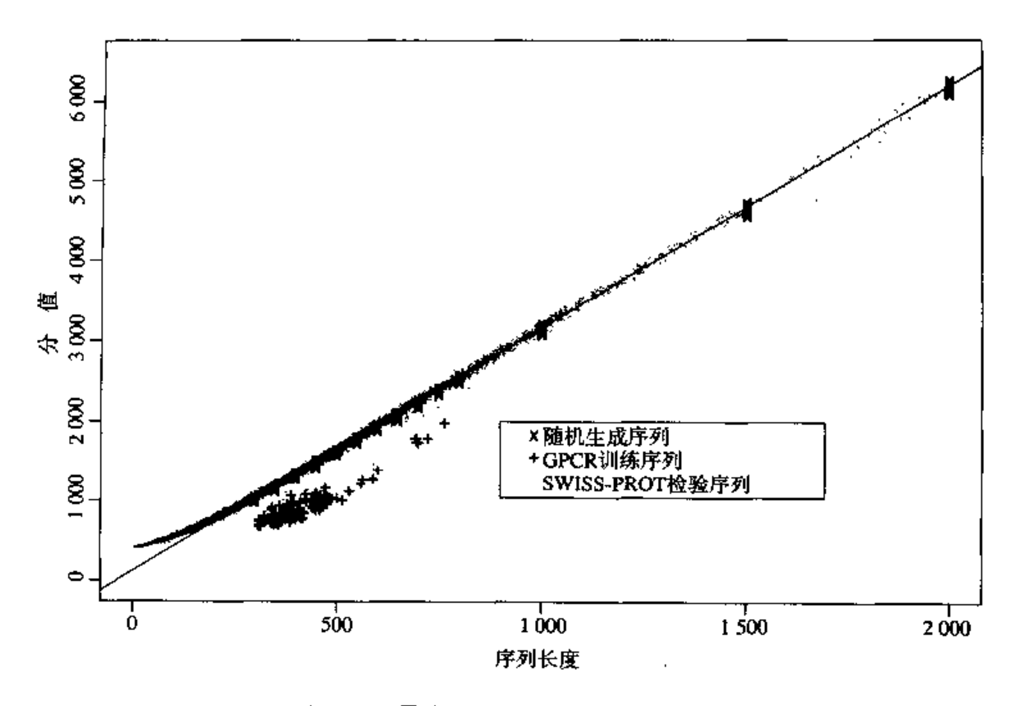


图8-2 最优Viterbi路径概率的负对数分值

图示序列包括142个GPCR训练序列，SWISS-PROT数据库中所有长度小于等于2 000的序列，以及220个平均成分组合与GPCR训练集相同的随机生成序列。这些序列分成长度为300、350、400、450、500、550、600、650、700、750、800的11组，每组各20个。回归线由220个随机序列拟合而得。

该模型可以清晰地鉴别平均成分组合相同的随机序列和真实的GPCR序列。与以前的实验[41,334]相一致，随机序列的分值和SWISS-PROT序列的分值沿着2条相似的直线聚集。沿某一直线聚集意味着在序列中添加1个氨基酸的平均代价近似为一常数。在很短的序列中，这一线性特征不再保持，因为它们对应那些不

正规的Viterbi路径（包含许多删除状态的路径）。而对于较长的序列（超过模型长度），这一线性特征的拟合精度随序列长度的增加而增加。因为对于平均成分组合固定的Viterbi路径，若其长度很大，则必定高度依赖于插入状态。而且实际上，系统在一些特定的插入状态上还被迫循环多次，这种情况在路径长度趋于无限大时将成为系统的主导行为。这些占主导地位的插入状态是代价最小的状态（概率最大）。很容易看到，对于特定的插入状态k，其代价依赖于两个同等重要的因素：自转移概率t_{kk}以及生成概率向量e_{kX}与序列的特定概率分布之间的互熵。更准确地讲，如果我们把由特定概率分布$P=(p_X)$生成的随机长序列的分值看做序列长度的函数，相应的分值将沿一条回归线聚集，其斜率为

$$\min_k[-\log t_{kk}-\sum_X p_X \log e_{kX}] \tag{8.1}$$

进而，对于较大的固定长度l，分值近似于正态分布（根据中心极限定理），方差为

$$l[\mathbf{E}_P(\log^2 e_{kX}-\mathbf{E}_P(\log e_{kX})]=l\mathbf{Var}_P[\log e_{kX}] \tag{8.2}$$

值得注意的是，分值的标准差随长度l的平方根增长。关于这些结果的证明以及更多的细节，见参考文献［38］。

表8-1　平均成分组合与GPCR训练集相同的随机序列的分值据［据（8.2）］

平均序列长度 l	序列数	实验平均分值	预测平均分值 3.038l+122.11	实验分值方差 0.66$\sqrt{l}$	预测分值方差
300	100	1 041.4	1 033.5	13.24	11.43
350	100	1 187.1	1 185.4	13.12	12.34
400	100	1 337.6	1 337.3	12.50	13.20
450	100	1 487.6	1 489.2	16.85	14.00
500	100	1 638.5	1 641.1	13.74	14.75
550	100	1 790.3	1 793.0	15.26	15.47
600	100	1 944.4	1 944.9	16.70	16.16
650	100	2 093.3	2 096.8	16.54	16.82
700	100	2 250.6	2 248.7	18.65	17.46
750	100	2 397.9	2 400.6	16.96	18.07
800	100	2 552.5	2 552.5	19.66	18.66
1 000	100	3 160.2	3 160.1	21.62	20.87
1 500	200	4 678.9	4 679.1	25.51	25.56
2 000	200	6 199.1	6 198.1	29.59	29.51

上述斜率公式是一个典型的渐近公式，它不一定适用于相对较短的序列，虽然实验中的大部分序列并不长。在上述模型中，对于平均成分组合与GPCR相同的序列，最优插入状态为第20号插入状态。实验得到的回归直线方程为

$y=3.038l+122.11$，而（8.1）给出的斜率预测值为3.039。由（8.2）估计，标准差按$\sigma\approx0.66\sqrt{l}$随长度增长。实验中得到的标准差与序列长度平方根之间的回归关系为$\sigma\approx0.63\sqrt{l}+1.22$。从表8-1中可以看到，理论估计和实验结果符合得很好。一般而言，预测精度会随长度增大而增大，这点对于标准差尤为明显。然而，在上述试验中，（8.1）和（8.2）对于长度与模型长度相当甚至短于模型长度的较短序列，仍然相当准确。对于平均成分组合与SWISS-PROT数据库相同的随机测试集，我们可以得到类似的试验结果。

8.1.4 分值归一化、数据库搜索和鉴别测试

在完成上述实验的统计分析后，我们开始研究如何进行鉴别测试这个基本问题，即如何通过算法决定某一给定序列是否属于该GPCR家族。一个很自然的想法是利用模型计算出的分值来鉴别GPCR序列和非GPCR序列。然而，我们并不能直接使用原始分值：首先，原始分值倾向于随序列长度增加而增加；其次，原始分值的离散程度在序列长度不同时差异较大，至少对于较长的随机生成的序列，其离散程度随长度的平方根增长。因此，我们需要先对原始分值进行中心化和标定。

这个归一化的过程可以通过几种方式完成。为了进行中心化，我们可以在不同长度上对实验结果进行平均，或者采用实验得到的回归线作为平均值，或者直接采用（8.1）和（8.2）估计的平均值。基于不同的目标，基准数据（base level）可以根据平均成分组合相似的随机序列或者根据某一真实数据库（如SWISS-PROT）中的序列计算获得。在上述实验中，这两者是相似的，但稍有不同。为了进行标定，我们可以采用实验得到的标准差或者理论估计的数值，它们同样可以由不同的数据源计算获得，如上述的SWISS-PROT数据库或成分组合相似的随机序列。由于每种方法都有其优缺点，因此在实际应用中，我们应该尝试多种不同的方法。一般来说，实验得到的估计更准确，但相应的代价也会大一些，特别是对于较长的序列，因为计算量$O(l^2)$随序列长度的平方增长。

利用真实的数据库进行中心化或标定时，如果在我们关心的长度区间内，数据库只能提供很少的序列，实验结果会有些问题；如果数据库中有些序列属于模型要预测家族，对此我们事先一无所知，因而没能把它们除去，实验结果同样也会有问题。尤其危险的是将这些数据引入标准差的估计。这里我们有必要采用一种迭代算法，其中每一步都必须根据前一步的计算结果剔除数据库中属于模型家族的一些序列，从而计算出一个新的标准差。这个新的标准差用于生成一组新的归一化分值，同时推断出属于模型家族的一组新的数据库序列。

另一个一般性的问题出在短序列上，因为短序列的行为经常有别于很长的序列。在某些问题中，对短序列要应用不同的归一化算法。最后，对于一个HMM模型库，平均成分组合与SWISS-PROT数据库相同的一组固定随机序列适用多个不同的模型。

在上述GPCR实验中，对于长度为l的任意序列O，我们用具有相似平均成分组合的随机序列的分值与实验得到的回归线的残差除以由（8.2）计算出的近似标准差作为归一化分值$\mathcal{E}_S(O)$:

$$\mathcal{E}_S(O)=\frac{[3.038l+122.11-\mathcal{E}(O)]}{0.66\sqrt{l}} \tag{8.3}$$

其中$\mathcal{E}(O)$是Viterbi路径概率的负对数。剩下的问题是如何选取鉴别阈值。这里，标识为UK33-HCMVA的训练序列取得最小分值16.03。这是一个孤立的分值，因为没有任何其他分值小于18。因此鉴别阈值可以设在16或更大一些。通过去除长度超过最大GPCR序列长度的超长序列以及包含歧义氨基酸的序列，基于上述归一化分值的搜索算法在阈值为16时，达到没有假阴性和两个假阳性的精度水平；在阈值为18时，达到一个假阴性和没有假阳性的精度水平。对于短序列（长度小于模型长度），（8.2）不一定是一个准确的近似，于是我们需要去尝试某种混合方案，例如用短序列（$l<N$）的实验结果计算出归一化系数，将（8.2）用于长序列（$l>N$）等。最后，一组固定长度的随机序列中的极端分值遵从一个极值分布，这一特征可以帮助我们选择阈值。[550]

8.1.5 亲水性图

由于GPCR家族带有特定的结构，一个合理的推论是：我们有可能根据某一公认的亲水性标度，通过绘制序列的亲水性图，轻易地鉴别某一给定的序列是否属于该家族。[166]如果真是这样，基于HMM的方法在鉴别实验中将不再那么重要，至少对于这个特定的家族而言。为了验证这一推论，我们为许多序列绘制了窗口宽度为20个氨基酸的亲水性图。图8-3给出了其中三个序列的例子。正如我们看到的，这些图中充满噪声和歧义氨基酸。因此，我们似乎不太可能仅仅通过亲水性图得到很好的识别率。保守模式、亲水性图和HMM应被视为一些互补的技术。

正如第7章所介绍的，我们可以根据HMM的概率分值绘制亲水性图。在这样的图中（如图8-4），显示的是每个位置上的亲水性期望值，而不是个别序列在特定位置上的亲水性观测值。结果，信号被放大了，7个跨膜区被清晰地标识出来。

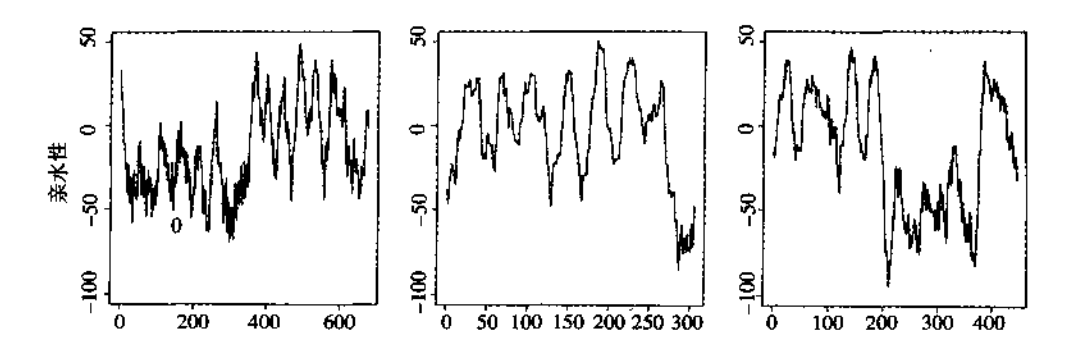


图8-3 长度小于1 000的三个GPCR序列的亲水性图（窗口宽度为20个氨基酸）

纵轴表示在特定的位置上，将一个假想的由20个氨基酸构成的α螺旋从膜内移动到膜外所需要的自由能。达到20 kcal/mol或更高的峰值通常预示着该位置可能存在一个跨膜的α螺旋。

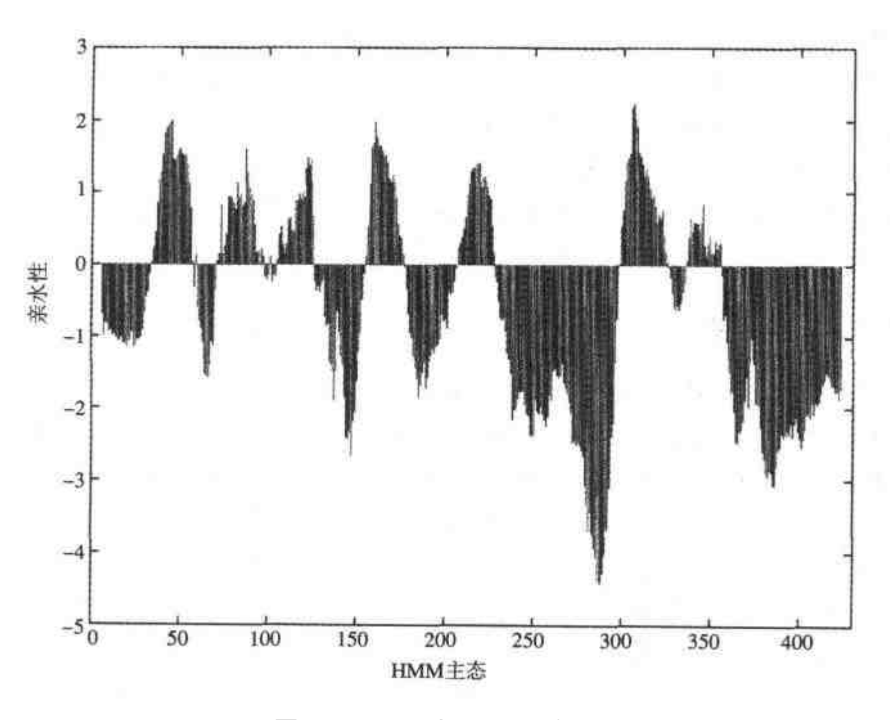


图8-4 GPCR相关HMM的亲水性图

8.1.6 细菌视紫红质

细菌视紫红质（bacteriorhodopsin）（简要综述见参考文献［317］，结构模型见［248］）是一种带有7个跨膜区域的蛋白质，它的功能类似于*Halobacterium*

*halobium*中的一种光驱动的质子泵（proton pump）。虽然其功能与视紫红质（rhodopsin）有关，但它并不是一种GPCR。细菌视紫红质和GPCR在结构和进化上的关系目前尚不完全清楚。对于参考文献［411］中给出的细菌视紫红质的原始序列，HMM计算得到的原始分值为852.27；而对于参考文献［318］中给出的稍有不同的序列，该原始分值为851.62。由于细菌视紫红质的序列长度l=248，这两个分值实际上非常接近于从平均成分组合相似的随机序列得到的回归线，只是稍低于该回归线。上述第一个序列的残差为23.26，由（8.3）算得其归一化分值为2.23。以上结果证实了细菌视紫红质不是一种GPCR，这与细菌视紫红质和GPCR之间缺乏显著的同源性这个结论相一致。

参考文献［414］中有一个观点：通过改变各螺旋之间的线性顺序可以得到更高的同源性水平，因此从进化的角度看，细菌视紫红质与GPCR可能通过外显子倒位彼此相关。于是，我们依照文中建议的方式，通过移动细菌视紫红质的7个螺旋并按（5, 6, 7, 2, 3, 4, 1）的顺序重新排列构造出一个新的序列。细胞外和细胞内的区域保持不变。这个人工序列的HMM原始分值为840.98。尽管这个分值比较接近于GPCR，但它与原始序列的分值并没有显著差异。由此看来，HMM的分值似乎不能对参考文献［414］中的假说提供足够有力的支持。由子细菌视紫红质的序列相对较短以及存在非螺旋区域，这些都可能对HMM的结果产生重要影响，因此在这方面我们还需要进一步研究。

8.1.7 分　类

“分类”是指将某一序列家族组织成一些亚族。分类是非常有用的，例如在系统进化的重构方面。利用HMM进行分类至少有以下两种方法：（1）采用竞争学习算法平行地训练多个模型（图8-5）；[334]（2）利用同一模型中概率和路径的聚类。第一种方法在这里不太适合，因为假设我们需要平行训练15个模型，现有训练集中的序列总数（尤其对某些受体亚族而言）是远远不够的。为了分类，我们需要开发更强大的算法，如在模型中包含更多的先验知识，同时还需要能够提供更多序列的新数据库。

对于第二种方法，通过观察可视化多重序列比对过程（对算法执行过程的逐步图示），可以清晰地看到：在对应于不同受体子类的Viterbi路径之间，存在聚集和插入关系。例如，在插入状态20上，所有促甲状腺素受体前体（thyrotropin receptor precursor, TSHR）都有一个很长的初始循环，该状态也是（8.1）的最优状态。有趣的是，同样的现象也存在于促黄体激素-促性腺激素受体前体（lutropin-gonadotropic hormone receptor precursor, LSHR）序列中。在这里，我

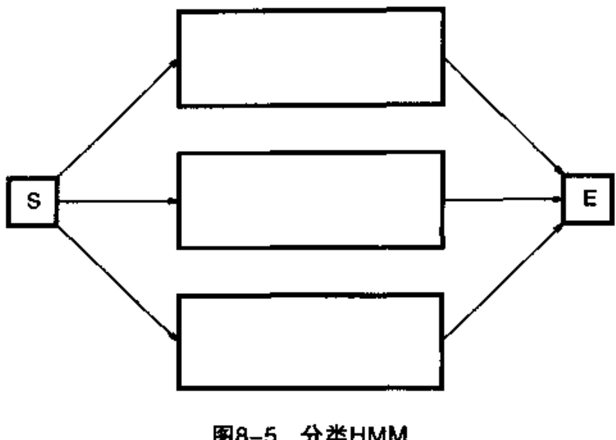


图8-5　分类HMM

这是参考文献［334］中用于蛋白质家族分类的多HMM构架示意图。在初始状态和终止状态间的每一个方框表示一个标准HMM构架。

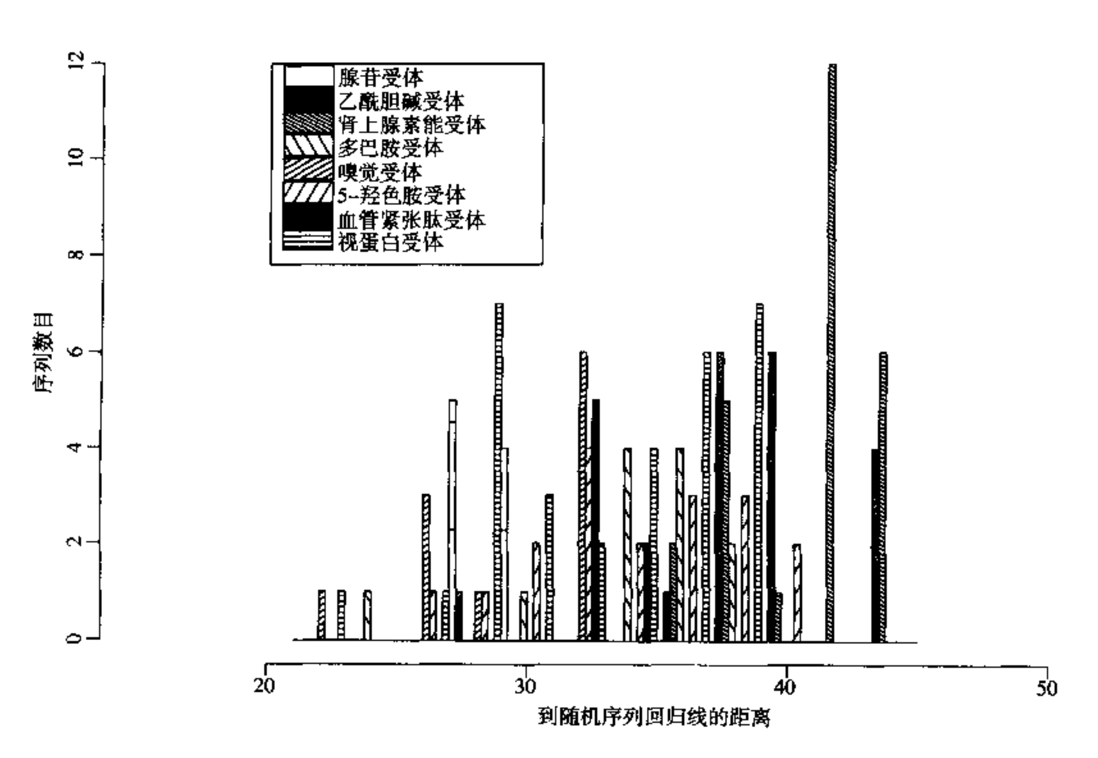


图8-6　不同类的GPCR序列到随机序列回归线的距离（归一化分值）直方图

嗅觉受体最接近于随机序列。肾上腺素能受体的分值呈现出最强的约束，并且距离回归线最远。不同类的受体倾向于在不同的距离上聚集。血管紧张肽受体的距离分布非常狭窄。

们并不想利用这类关系系统化地对GPCR进行分类，我们只想根据受体主要亚族的已有分类，研究HMM分值（在算法执行过程中）的特点。

针对这一目的，我们首先对所有受体类别抽取至少7个有代表性的序列，以避免主体偏倚（major bias effect）。受体的分类和各类中相应的序列数量为：嗅觉（olfactory）（11个）、腺苷（adenosine）（9个）、视蛋白（opsin）（31个）、5-羟色胺（serotonin）（18个）、血管紧张肽（angiotensin）（7个）、多巴胺（dopamine）（12个）、乙酰胆碱（acetylcholine）（18个）、肾上腺素能受体（adrenergic）（26个），共计132个序列，代表了从SWISS-PROT中搜索所得的扩展GPCR数据库中62%的序列。图8-6是按以上方法选择的8类受体序列相对于随机回归线的距离（或归一化分值）的直方图。归一化分值的取值范围为20~44，直方图的柱体宽度为2。

给定受体亚族中，序列在某些特定的归一化分值附近聚集的现象是非常引人注目的。嗅觉受体最接近于随机序列，这并不奇怪，因为这类受体需要与很多可能的气味相互作用。而肾上腺素能受体则距随机回归线最远，因而表现为受约束最强的一类受体。每类受体的标准差也有显著差异。例如，血管紧张肽受体仅在一个窄带上取值，并且只有一种已知类型，而视蛋白受体的取值范围则宽得多。大多数亚族的直方图呈钟形分布，但也有例外。视蛋白表现为双模态（bimodal）分布，这可能是由于视蛋白受体亚族中还存在更小的子类。第二个峰值绝大部分对应于视紫红质（OPSD）序列，还有少数对应于红色敏感视蛋白（red-sensitive opsins, OPSR）。这两个峰值的存在似乎不是由脊椎动物视蛋白和无脊椎动物视蛋白之间的差异引起的。在未来的数据库中，有可能进一步提高分辨率并降低采样误差的影响。然而，以上结果已经足以揭示：基于HMM的序列分值与序列隶属于给定受体亚族之间有很强的相关性。另一方面，值得注意的是，单纯从分值的直方图出发，在不引入任何关于受体类型的先验知识的情况下，我们很难发现隐藏在序列后面的分类结构。整个GPCR家族的详尽分类以及完整的系统进化重构已超出本章的讨论范围。

8.1.8 EST和cDNA中的片断发现

作为过去几年中EST和cDNA测序工作的结果，已经有一些对应于蛋白质片断的DNA序列数据库。于是我们自然会对识别和分类这些片断，以及从中发现一些新信息感兴趣。HMM可以从多个方面适应这些工作。一种显然的可能性是，对于给定的蛋白质家族，我们可以训练不同的模型以识别蛋白质的不同部分。这里我们用GPCR家族和人工生成的片断进行一些初步的测试。假设我们感兴趣的

片断长度为l=150，我们会同时考察更短的序列。另外，我们将序列噪声也纳入考察范围。测序中的噪声主要来源于将氨基酸转换为DNA时引入的随机独立的碱基改变，可以将其近似为固定的噪声概率p。我们将集中考察三个长度等级：l=150，100和50，以及三个不同的噪声水平：p=0%，5%和10%。

我们首先构造5个数据集，它们都只包含长度为150的片断。第一个数据集是从GPCR训练集的142个序列中按随机位置抽取的片断。第二个数据集的200个片断是从一个更大的GPCR数据库中随机抽取的。[325]第三个数据集包含随机生成的200个长度为150的片断，其平均成分组合与GPCR相同。第四个数据集包含从激酶序列数据库中随机抽取的长度为150的片断。第五个数据集用与第四个数据集同样的方法从SWISS-PROT数据库中抽取。

与逐项比对类似，HMM可用于生成局部比对和全局比对。这里我们分析与模型全局比对相关的分值，即完整Viterbi路径的概率的负对数。图8-7绘出了相应分值的直方图。特别值得注意的是，这些结果显示，在原始分值的阈值为625时，搜索结果很好地消除了假阳性，同时只有少量的假阴性出现。图8-8显示了相同的结果，只是长度为l=50、噪声水平为p=10%。正如我们所看到的，分布的重叠部分变得更加显著。这要求我们对片断长度以及分布于整个SWISS-PROT数据库的噪声对精度恶化的影响进行更加精确的分析。

实验结果的总结

图8-9总结了全部实验结果。横轴用于显示片断长度，纵轴用于显示片断的分值。图中同时描绘出了目标序列（GPCR）和非目标序列的标准差（竖条）以及分值的取值范围（细线），其中包括所有片断长度（50，100，150）和所有噪声水平（0%，5%，10%）。对于每个片断长度，细线表示所有噪声水平上的目标和非目标序列的分值的取值范围。为了能够表达所有噪声水平上的所有取值范围，图中代表分值范围的长度坐标位置与片断的实际长度坐标略有偏移。

对于某一给定的片断长度（例如50），六条线从左向右分别代表：噪声水平0%的目标片断、噪声水平0%的非目标片断、噪声水平5%的目标片断、噪声水平5%的非目标片断、噪声水平10%的目标片断、噪声水平10%的非目标片断。我们可以针对目标和非目标片断计算各个噪声水平上的回归线：

- 目标片断

　　噪声水平0%：　$y=387.4+1.199l$

　　噪声水平5%：　$y=384.0+1.314l$

　　噪声水平10%：　$y=382.3+1.401l$

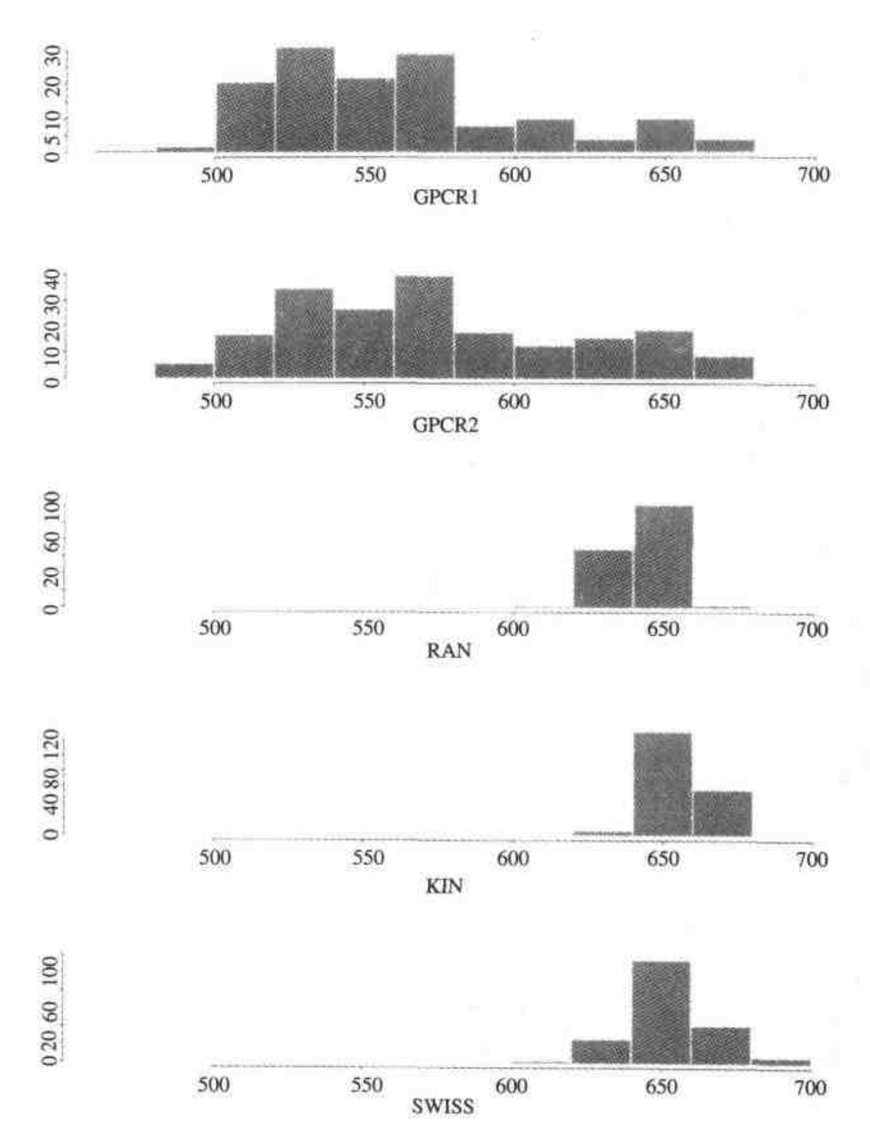


图8-7 长度为150的不同片断的分值直方图

第一个直方图基于从GPCR训练集中抽取的142个随机片断。所有其他直方图都基于以随机方式抽取的200个片断，它们分别来源于：一个更大的GPCR数据库、用相似的平均成分组合随机生成的片断、一个激酶数据库以及SWISS-PROT数据库。

- 非目标片断

  噪声水平0%：　　$y=364.7+1.909l$

  噪声水平5%：　　$y=364.8+1.910l$

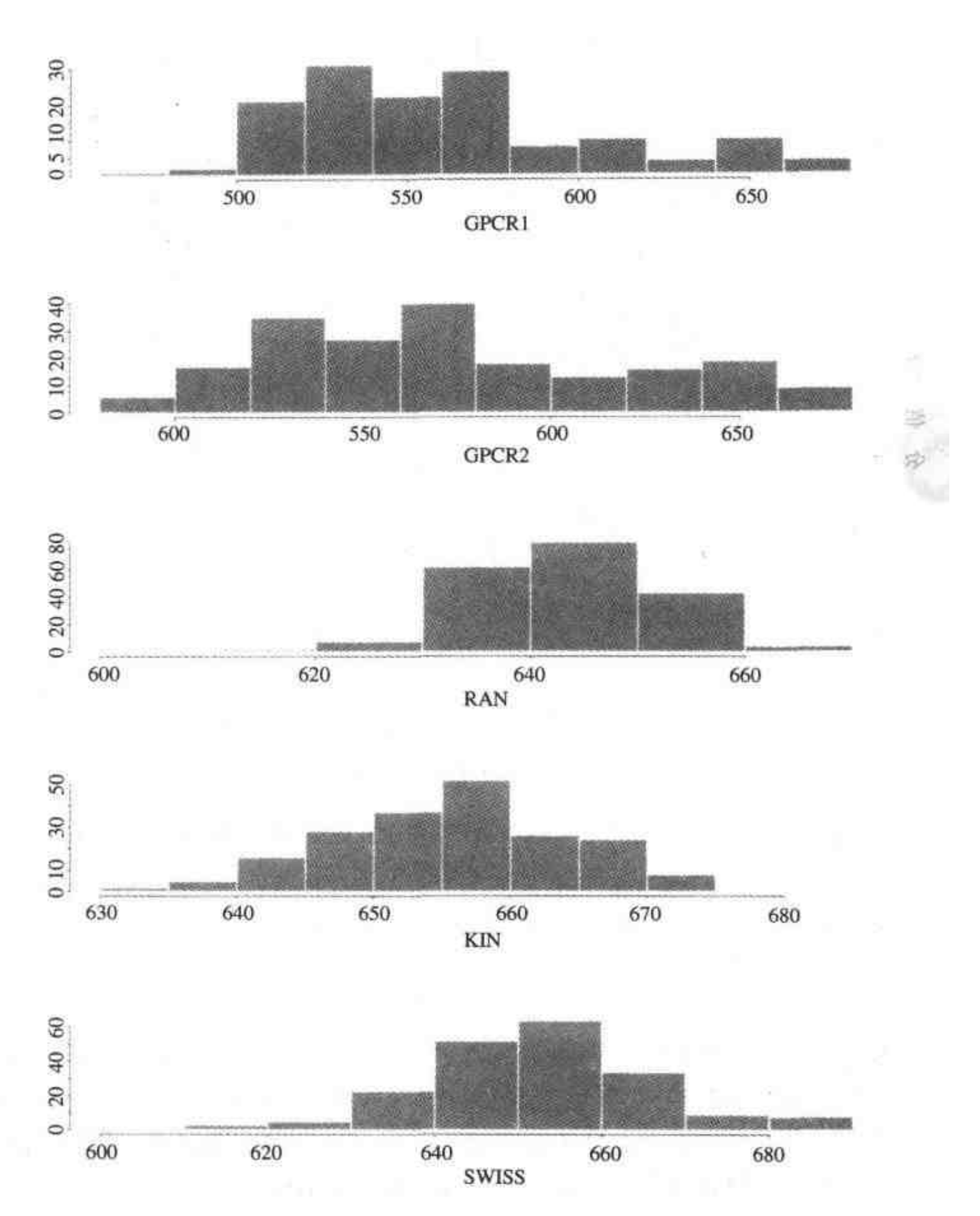


图8-8　长度为50，噪声水平p=10%的片断的分值直方图

噪声水平10%：　　$y=364.8+1.911l$

这些回归线仅仅由三个长度上的片断分值确定，因此只能作为其他长度片断分值的某种近似。这些回归线在片断长度约为35处相交，这意味着仅仅基于分值的有效鉴别的长度下限约为35。

正如我们所预期的，目标序列回归线的斜率随噪声水平的提高而增大，交点

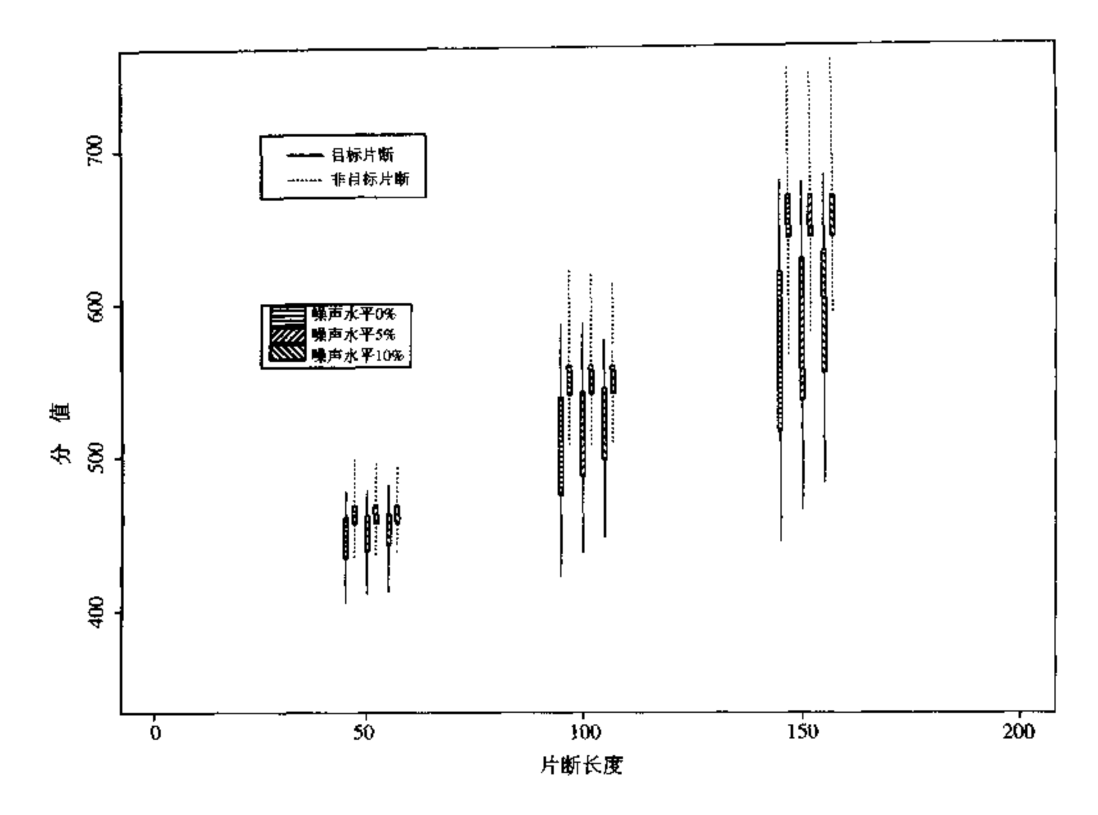


图8-9 全部SWISS-PROT数据库片断分值的总体分布

横轴用于显示片断的长度，纵轴用于显示片断的分值。图中同时描绘目标序列（GPCR）和非目标序列的标准差（竖条）和分值的取值范围（细线），其中包括所有片断长度（50，100，150）和所有噪声水平（0%，5%，10%）。

变化不显著。非目标序列的斜率和交点都很稳定，噪声水平对非目标序列的影响不大。全部非目标序列的近似回归线为$y \approx 364.8+1.91l$。与参考文献［38］中的结果一致，这一斜率略低于从更长的序列导出的回归线斜率。我们可以用类似的方法研究分值的标准差作为长度和噪声水平的函数。

ROC曲线

在计算出数据库中所有相关片断的分值之后，我们可以进一步统计在每个长度和每个噪声水平上，在任意给定分值阈值的条件下，真、假阴性和真、假阳性的数量。这些敏感性/选择性结果可以绘制成相应的ROC曲线，如图8-10所示。

ROC曲线是通过在给定范围内扫描阈值，根据相应的真、假阳性和真、假阴性的个数，计算敏感性或命中率（真阳性的比例）以及选择性或误报率（假阳性

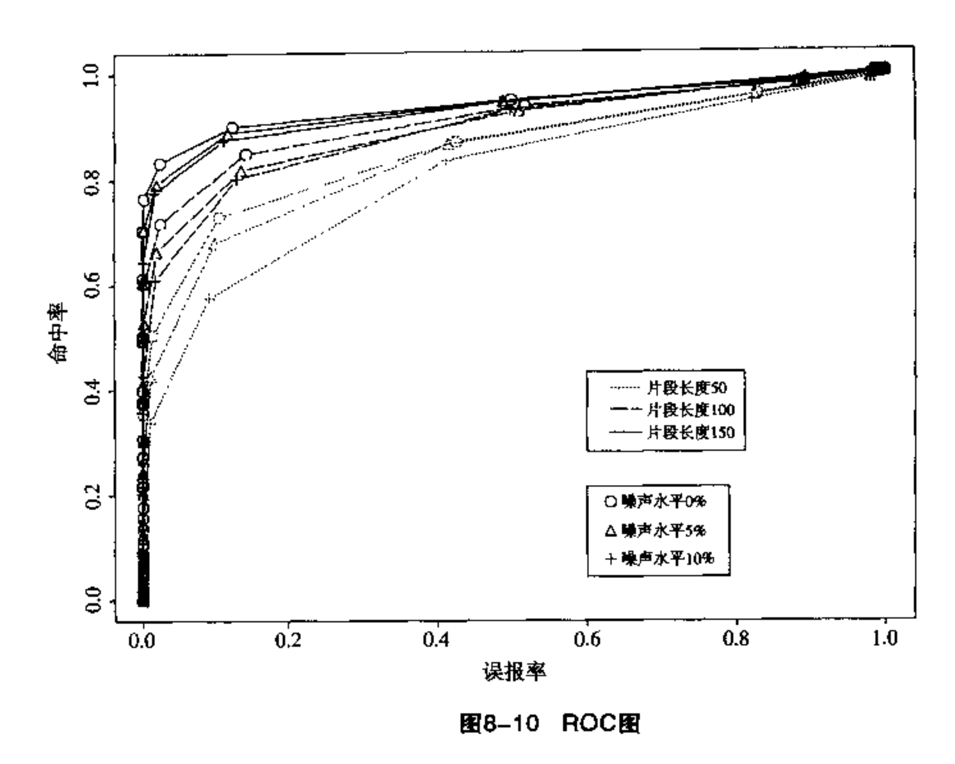


图8-10 ROC图

SWISS-PROT数据库中所有长度为50、100或150的序列，在噪声水平为0%、5%和10%的所有分值的ROC图。这些序列中不包括含有歧义符号的序列。

的比例）。阈值范围是片断长度的函数。对于每个片断长度，最小阈值（范围）应该保证在所有噪声水平上没有非GPCR片断在分类中呈阳性；最大阈值范围则应该保证在所有噪声水平上没有已知的GPCR（来源于PROSITE数据库）在分类中呈阴性。这些曲线为我们根据不同目标选择阈值提供了一个方便的手段。如图8-10所示，ROC曲线作为噪声水平和片断长度的函数时，曲线族呈现出显著的渐近排列特征。图中曲线倾向于贴近纵轴，这意味着即使在命中率较高时，仍然可以得到很低的误报率，即对大多数目标序列有很好的识别效果。然而，图中曲线并不倾向于贴近横轴，这意味着为了达到更高的目标序列识别比率，假阳性的数量就会显著增加。这当然因为GPCR是由相对保守的区域和高度可变的区域共同构成的。我们几乎不可能在一般性的SWISS-PROT数据库背景上，有效识别出一个取自高度可变区域的相对较短的片断。同样，包含更加保守区域的相对较长的片断则比较容易从背景中分离出来。当片断长度较短以及噪声水平较高时，这些曲线提示我们可以通过构建附加的过滤器提高性能。

d' 测度与识别结果分析

给定两个被鉴别群体的分值，并假设这两个分值的分布都为标准差为1的高斯分布，对于特定的假阳性和假阴性水平，d' 测度给出两个高斯分布中心之间的距离。

利用d' 测度对SWISS-PROT数据库识别结果的初步分析显示，对于不同的阈值，d' 测度的变化幅度很大。这表明分值的分布曲线并不是高斯分布（正如我们从直方图中看到的）。由于对不同的噪声水平和片断长度给出一种统一的性能评价方法很有意义，我们仍然使用d' 测度分析识别。对不同噪声水平和片断长度，我们对命中率为0.9时的误报率进行线性插值，并对获得的数据对（0.9，x）计算d' 测度，其中x为线性插值结果。表8–2给出针对不同噪声水平和片断长度的d' 测度结果。

表8–2 根据全部实验结果，对不同片断长度和噪声水平上的分类性能的初步评价

	0%	5%	10%
50	1.16	1.18	1.03
100	1.63	1.49	1.50
150	2.41	2.14	1.96

提高识别率

到目前为止我们只考察了HMM产生的原始分值，即Viterbi路径概率的负对数。然而理论上，HMM包含更多的信息，可以用于提高数据库挖掘性能。事实上，对于每个片断，可以建立更多的标识，将它们组合使用以提高性能，例如在一个贝叶斯网络中就可以考虑使用这种方法。最值得注意的是可以利用路径本身的结构。正如我们预期的，真阳性和假阳性的路径有显著区别。假阳性的路径一般更不连接并包含许多间隙。于是，我们可以建立一些路径非连续性的指标，这些指标包括：（1）路径上始于删除状态的转移的次数；（2）路径上生成状态的最长连续片断的长度；（3）路径本身的概率的对数（只包括转移，不包括生成）。在一项测试中，将这类指标与原始分值结合可以将识别的命中率提高15%～20%。参考文献［42］中探讨了提高识别率的其他研究思路。

8.1.9 HMM用于信号肽和信号锚的预测

在6.4.1节中，我们介绍了在原核生物和真核生物序列的N端发现信号肽的问题。基于窗口的神经网络方法[404]可以利用氨基酸之间的相互关系，尤其是在剪切位点附近。然而在没有附加输入单元的情况下，这种神经网络方法不能从序列

的整体模式以及信号肽特有的不同长度分布中获益。

事实上，我们已知信号肽的长度特征在不同类型的组织中有所不同：细菌的信号肽长于相应的真核生物的信号肽，而革兰氏阳性菌的信号肽要长于革兰氏阴性菌的信号肽。另外，随着在信号肽中位置的变化，对应的成分组成也有所不同，类似的情况也存在于成熟蛋白的前几个残基中。

另外一个重要的难题是某些蛋白质的N端带有与信号肽相同的初始移位序列，然而这些序列并没有被肽酶（peptidase）剪切。[541,406] 没有被剪切的信号肽被称为信号锚，它是一类特别的膜蛋白。信号锚通常比剪切过的信号肽包含更长的疏水区，其他区域的成分组合特征也与信号肽不同。

尼尔森和克罗建立了一个HMM模型，它不仅能够鉴别信号肽和非信号肽，而且能够确定剪切位点。[406] 该模型在设计中考虑了已知的信号肽特性，尤其是6.4.1节中介绍的不同的区域。为了获得能够分辨信号肽和信号锚的预测工具，他们的模型构架将一个信号肽模型和一个信号锚模型结合在一起。

信号肽模型见图8-11。利用“捆绑”状态（tied state）对不同区域中的长度分布进行显式建模，“捆绑”状态的生成和转移概率有相同的氨基酸分布。

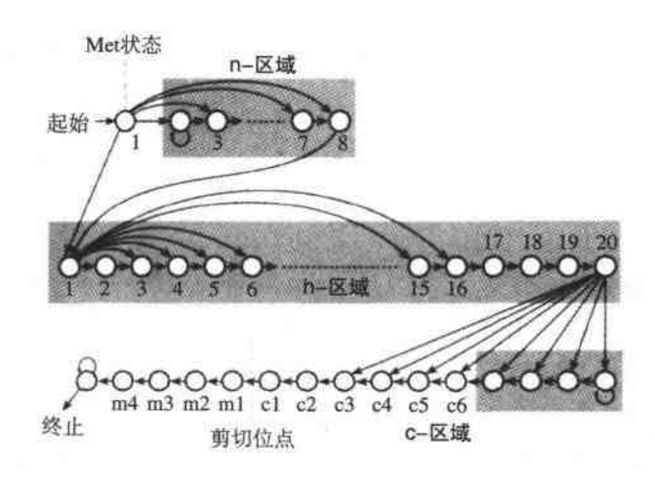
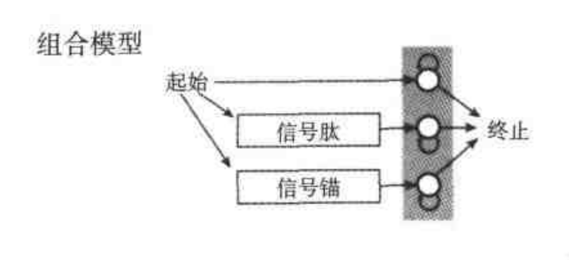


图8-11 用于信号肽鉴别的HMM

模型[406]的设计实现了对不同区域的长度分布的显式建模。阴影区域的状态相互“捆绑”在一起。

为了鉴别信号肽、信号锚和水溶性的非分泌蛋白，模型中增加了一个图8-12所示的信号锚模型。整个模型通过各类序列训练获得（包括已知的信号肽、已知的信号锚以及其他细胞质和细胞核序列）。序列在组合模型中的最佳路径能够预测该蛋白质属于这三类蛋白质中哪一类。



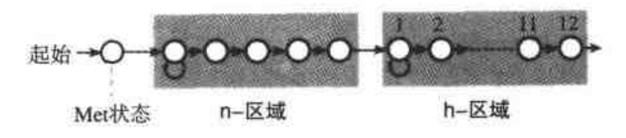


图8-12 设计用于鉴别信号肽和信号锚的HMM

上方的方框图显示组合模型[406]如何将信号肽模型和信号锚模型结合在一起。阴影部分的状态彼此捆绑在一起，用于所有不在信号肽或信号锚中的残基建模。下方为信号锚的模型，它只包含两类状态（用两个阴影框分开），并与Met状态分离。

就鉴别信号肽和非信号肽的预测性能而言，结合C值和S值的神经网络（见6.4.1节）可以达到与HMM相当的鉴别精度。神经网络对于真核生物序列的效果稍好，而HMM对于革兰氏阴性菌的效果稍好。[406]对于鉴别剪切后的信号肽和没有剪切的信号锚，上述HMM的相关系数达到0.74，对应的敏感性为71%，特异性（选择性）为81%；而由神经网络的S值获得的相应性能评价中，相关系数不超过0.4。因此在识别信号锚，进而在识别同类的膜结合蛋白（membrane-associated protein）的应用中，HMM有显著的优势。

然而，我们并不能就此断言神经网络不适用于信号锚识别问题，因为与上述HMM不同，神经网络的训练集中并未包含信号锚序列。[406]

在TMHMM方法中，类似的构建特定结构HMM的方法被用于建构和预测跨膜蛋白质的拓扑结构。[335]TMHMM方法能够准确地鉴别水溶性的蛋白质和膜蛋白，其特异性和敏感性可同时达到99%以上，但当有信号肽出现时精度会下降。由于TMHMM方法的精度非常高，它非常适用于扫描整个基因组以发现整合性膜蛋白（integral membrane protein）。[335]

8.2 在DNA和RNA方面的应用

核苷酸序列的多重序列比对比蛋白质序列的多重序列比对更困难。其中一个

原因是氨基酸替换矩阵中的参数可以通过进化和生化分析进行估计，然而估计一般性突变以及从核酸中删除单个核苷酸的代价则要困难得多。对于字符集较小的序列，比对的显著性将更快地达到不确定的“边缘地带”（twilight zone），因而在进行DNA比对时，进入边缘地带所需要的进化事件（核苷酸的插入或删除）更少。

HMM不要求对替换代价（substitution cost）给出先验的显式定义。通过将多对多方式转化为多对一的方式，HMM有效避免了传统多重序列比对方法在计算上的困难。[155]模型中不同的位置在实际中分别对应于隐含的替换代价。在有关核酸的一些应用中，HMM的这些特性已成功地指导我们发现了一些其他方法尚未揭示的新模式。而在与蛋白质相关的应用中，HMM带来对早期方法的更多改进。

8.2.1　原核生物和真核生物的基因发现

基因发现要求许多不同信号的集成：启动子区域、翻译起始和终止的上下文序列、阅读框的周期、聚腺苷酸化（polyadenylation）信号；对于真核生物还包括：内含子剪接信号、外显子和内含子的成分对照、核小体定位的潜在差异及序列拓扑区域的决定子（sequence determinant）。其中最后一类信号涉及间质连接区（matrix/scaffold attachment region，简称MAR或SAR），它与染色体的高级组织结构有关。连接信号可能涉及有机体内部的启动转录行为，近来已有它们在基因之间出现的相关报道。对于原核生物，DNA序列还要能被压缩成很紧凑的类染色质（chromatin-like）结构。从一个单一操纵子（operon）扩展的DNA的长度对应于细胞的直径。由于所有这些信号在很大范围内彼此互补，在一定程度上，某些信号较弱时，另一些信号可能较强，因此用概率的方式将它们集成起来，是对付这类复杂性的一种很自然的办法。

在原核生物中，编码区不会被间插序列（intervening sequence）打断这一事实使基因发现变得更简单。然而，区分代表真正基因的序列和不代表任何基因的序列并不容易，尤其是当开放的阅读框相对较短时。在非常成功的基因发现程序GeneMark[81,83,82]中（它的第1版基于片断相关的非均一结构的马尔可夫模型），一个显著提高性能的关键特性是它能够聪明地识别一个真正的编码区在非编码链（non-coding strand）上留下的“影子”（更多的细节参见第9章）。

人们已经开发出一种能够在大肠杆菌DNA中发现蛋白质编码基因的HMM（在这个模型开发出来时，大肠杆菌全基因组测序尚未完成）。[336]该HMM中包括对大肠杆菌基因的编码子及其序列建模的状态，以及对在基因间区域中发现的

模式建模的状态。这些模式包括重复的基因外回文序列（extragenic palindromic sequence）和夏因—达尔加诺motif（一段具有特定功能的生物序列）。考虑到原始DNA序列中潜在的测序错误以及移码突变（frame shift），HMM允许在编码区出现单个核苷酸的插入和删除（尽管可能性很小）。该HMM的参数估计使用了标注DNA的大约100万个核苷酸，并在包含约325 000个核苷酸的不连接的片断集合上进行测试。这个HMM发现了大约80%的已知大肠杆菌基因的精确位置，以及约10%的大概位置。它还发现了一些潜在的新基因，并且定位了一些出现在片断中的插入/删除错误或移码突变。

在真核生物的基因发现方面，人们已开发出许多强大的HMM和其他概率模型（参见第9章和参考文献［343,107］以其中的相关参考文献）。典型的真核生物基因模型通常由多个子模型组合而成，如剪接位点子模型、内含子子模型以及外显子子模型等，以便利用其信号一致性较弱和成分不同等特点。如果模型目标是在有限的时间内扫描整个基因组，单个子模型的规模就必须保持相对较小。其他关键要素包括：考虑到内含子打断阅读框有三种可能方式，需要并行地应用三个外显子子模型；配合利用外显子和内含子的长度分布、启动子、聚腺苷酸化信号、基因间序列和链的对称性等特性。通常将整个识别系统一次训练完成比分别训练各个子模型效果更好。特别地，为了使系统产生整体最优的基因分析而不仅是最优的序列概率，我们可以对标准HMM算法进行改进。[333] 这些模型[107]的最佳基因识别水平可以达到75%～80%的外显子完全识别率（包括准确的剪接位点）。进一步提高识别率还需要更多的工作。新的改进可能来源于引入新的和更好的子模型（如启动子或起始和终止外显子的子模型），以及DNA中的其他物理特性和信号（如可弯曲性或核小体定位等）。这些至今仍被人们完全忽视的精确信号，很可能在生物的基因发现机制中扮演重要的角色。在本章后半部分中，我们将利用不同的子模型构造一些较大的基因模型，并描述这些可能的信号。

8.2.2 人类基因剪接位点、外显子和内含子的HMM

在mRNA离开核酸被翻译为蛋白质之前，真核生物基因中包含的间插序列（内含子）从mRNA分子中被剪去的机制称为剪接。至今已有大量的研究投向理解内含子剪接的分子机制。由于人们尚很不清楚特定剪接所必需的和足够的序列决定子，所以HMM形式的概率模型被用来描述实验发现的剪接信号。

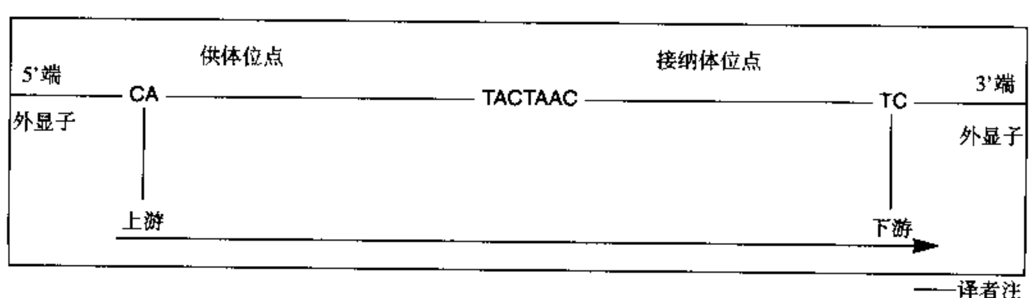
与蛋白质家族的情况不同，DNA有一个非常重要的特点值得注意，即所有的外显子及其相关的剪接衔接点（splice site junction）在进化上的相关性既不直接也不紧密。然而，就一般性的共同特征而言，它们仍然构成一个“家族”。例如，

在一组旁侧扩展的外显子（flanked exon）序列的多重序列比对中，剪接位点的保守序列会作为模型中的高度保守区而突现出来，就像蛋白质家族中的一个蛋白质motif（超二级结构模体）。因此，我们应该特别谨慎地将HMM中的插入和删除状态视为一般的字符串操作，而不是进化事件。

为了测试HMM能否很容易地发现人类DNA中接纳体位点和供体位点的已知特性，我们用1 000个随机选择的旁侧扩展的供体和接纳体位点序列，训练了一个如图7-2的标准构架模型。[32,33,35]通过仔细检查HMM在旁侧扩展的接纳体位点训练得到的参数，我们可以看到模型精确地获得了接纳体的保守序列：（[TC]…[TC] [N] [CT] [A] [G] [G]）。图8-13中，碱基文本清晰可见，同时还可以看到其他一些已知的弱信号，如分支［套索式（lariat）］信号在内含子的3'端有一个出现概率很高的A。

类似地，从旁侧扩展的供体位点序列训练获得的模型中，可以清晰地看到供体位点，只是比接纳体位点难以学习。模型同样精确地获得了供体的保守序列：（[CA] [A] [G] [G] [T] [AG] [A] [G]）。对于从供体位点向下游扩展了约75个碱基（见图8-13）的G-rich区，该模型同样有效。对于接纳体位点更容易学习这一现象，最可能的解释是接纳体位点的延展特性更好，而供体位点则较差。然而，也可能由于训练序列中的外显子总是固定向上游旁侧扩展100个核苷酸。为了检验这一假设，我们用同样的序列训练了一个类似的模型，只是采用相反的顺序（从右向左）。出乎我们意料的是，新模型对于接纳体位点的学习效果仍然显著优于供体位点（在新模型中，供体位点处于接纳体位点的下游）。在碱基文本区（固定长度为175个碱基）处于接纳体位点下游的部分（外显子左端）中，核苷酸排列有较高的随机程度，这种随机性可能促成了以上学习结果。相反的，处于内含子5'端的G-rich区具有某种全局性的结构，能够被HMM识别，使得供体保守序列的特征难以突显出来。[⑥]

⑥ 本节关于剪接模型的论述比较抽象，特补充以下剪接示意图：



——译者注

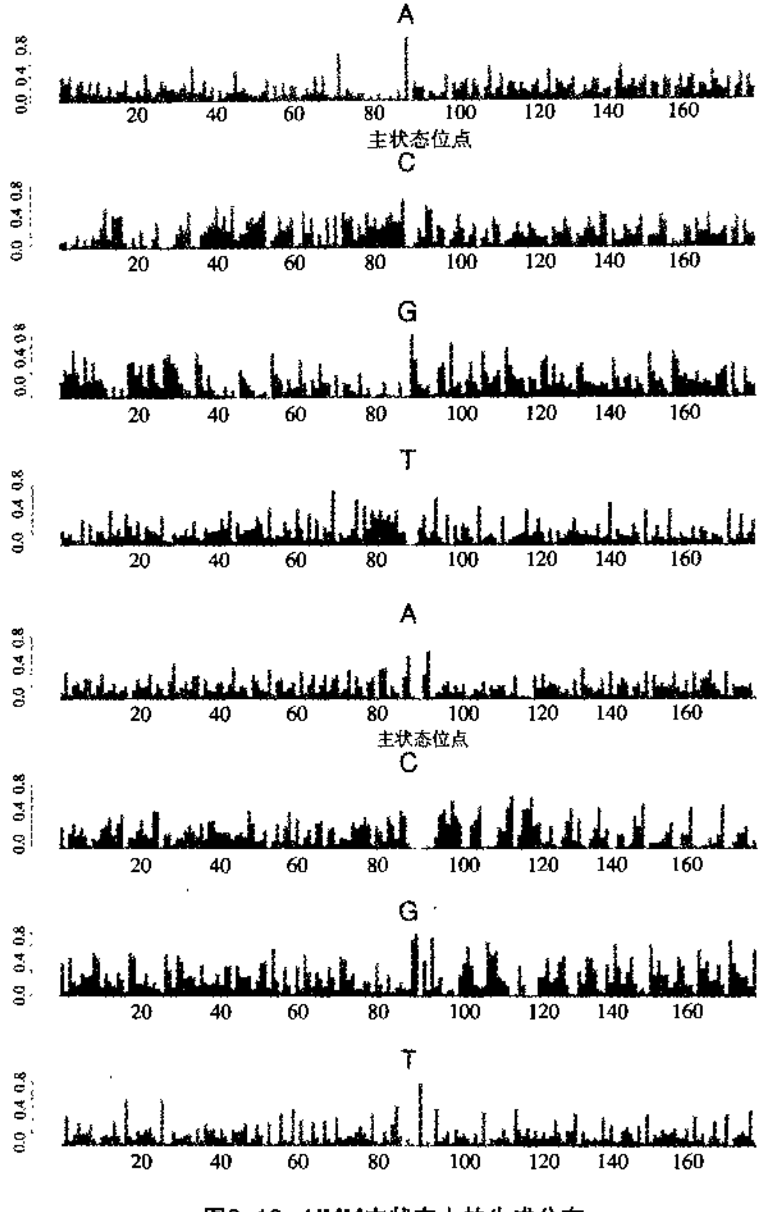


图8-13 HMM主状态上的生成分布

模型的训练集分别为：1 000个接纳体位点（顶图）和1 000个供体位点（底图）。每个位点的旁侧序列保持定长为100个核苷酸，而模型长度为175。对于接纳体位点，很容易识别出特征保守序列（[TC]…[TC] [N] [CT] [A] [G] [G]）。注意在接纳体位点的下游与分支点相应的位置上，碱基A的频率很高。供体位点的特征保守序列也很容易识别（[CA] [A] [G] [G] [T] [AG] [A] [G]）。训练过程用标准构架初始化（见图7-2），并在目标函数中添加偏向于主干转移路径的正则项。

8.2.3　利用HMM新构架在外显子和内含子中发现周期模式

在另一组实验中，我们用包括内含子序列的旁侧扩展的人类外显子序列训练一个标准构架的HMM。训练集为随机选择的旁侧扩展的内部外显子，外显子的长度限制为100~200个核苷酸（人类内部外显子平均长度约为150个核苷酸）。

沿着模型伸展方向主状态，4种核苷酸各自的生成概率见图8-14。从图中可以看到引人注目的周期模式，特别是在外显子区域，其特征是最小周期为10个核苷酸，A和G的相位相同、C和T的相位相反。模型的参数在以下位置有形式为[AT] [CG]（或[AT]G）、周期约为10个碱基对的周期模式：10，19，28，37，46，55，72，81，90，99，105，114，123，132，141。对模型主干上的生成谱也可进行2个核苷酸的联合比较。A+G和C+T的曲线与A+T和C+G的曲线相比，无论

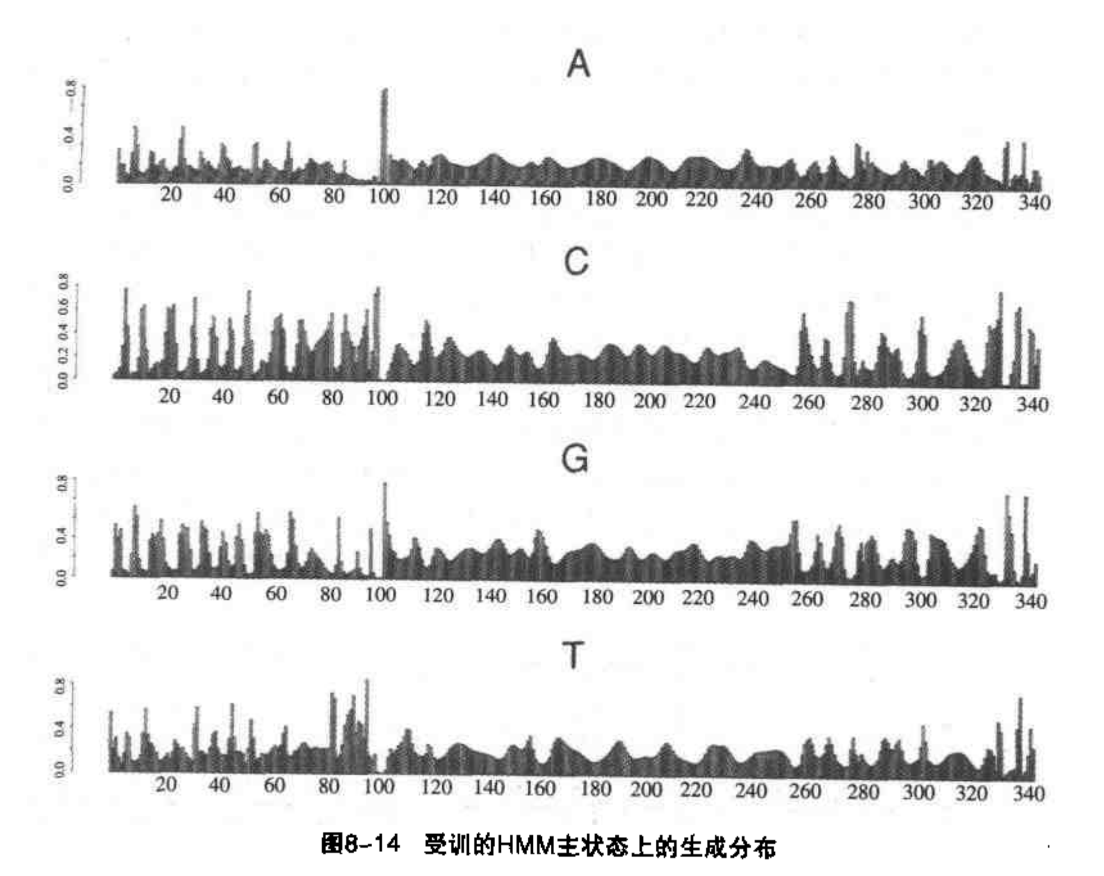


图8-14　受训的HMM主状态上的生成分布

训练集为500个旁侧扩展的内部外显子。外显子的长度限制在100到200个核苷酸之间，平均为142个，两个旁侧的内显子序列长度都固定为100个。模型包含342个主状态。注意外显子区域和旁侧区域在振幅模式上的差异。

在内含子一侧还是在外显子一侧都要平滑得多。无论对于A+G和C+T的平滑同相/反相模式，还是对于A+G在前、C+T在后的沿相反方向急剧变化的模式，都可以看到10个核苷酸的周期性。还有一种大约包含3个碱基对的周期性，在C+G曲线中尤其明显，其中每3个生成状态对应1个局部最小。这与人类基因的阅读框特性相一致，[525]它在第三个密码子的位置上特别强（C的出现概率约为30%，G的出现概率约为26%，见图6–11）。

为了进一步描述这一个周期特征，我们针对非旁侧扩展的内部外显子训练了一系列不同的HMM构架，以便从出现在起始和终止外显子上的核苷酸组成的特殊梯度中提取周期特性。[164]无论采用什么构架，当训练集为大量长度在100到200个核苷酸的内部外显子时，在生成概率中清晰地涌现出一致的模式。这些不同的构架包括传统的从左到右模型，相同的片断被捆绑在一起的从左到右模型在噪声存在时仍能很好地揭示周期模式的环状或轮状模型。尽管传统的从左到右的构架并不是外显子的理想模型，但由于外显子长度变化很大，它仍然能相当好地识别出周期模式。

为了进一步测试周期性，我们训练了一个带有周期为10的硬连接（hardwired）的“捆绑”外显子模型。[33]这个捆绑模型由14个长度为10的相同片断以及模型首尾各5个附加位置构成，模型总长度为150。在训练过程中，通过参数的“捆绑”使片断之间保持一致，即在训练中强迫参数在片断间保持一致，正如神经网络中的权重共享过程。模型用800个长度为100~200个核苷酸之间的内部外显子训练，并用262个不同的序列进行测试。图8–15显示了训练得到的重复片断的参数。生成概率用与长度成比例的水平短线表示，这个片断中包含许多结构。其中最主要的特性是在位置12~14上的正则表达式[^ T] [AT]G。这一模式经常出现在上述标准模型中信息熵很低的位置。为了测试显著性，我们将捆绑模型与长度相同的标准模型相比较。通过比较两个模型在外显子序列和成分相似的随机序列上的负对

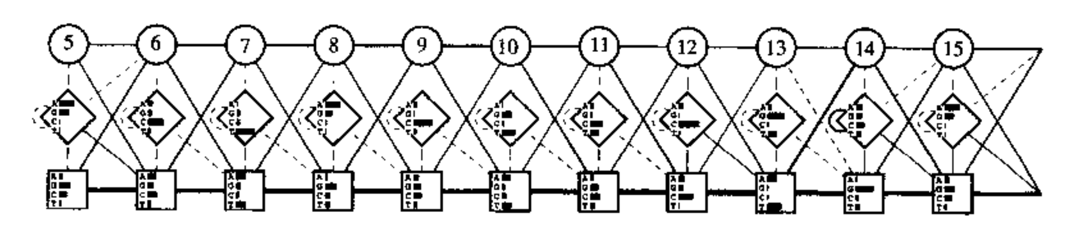


图8–15 捆绑模型的重复片断

长方形表示主状态，圆形表示删除状态。直方图表示主状态和插入状态的生成分布。连接线的宽度与转移概率分布成比例。位置15与位置5相同。

数概率的平均值，可以清楚地看到捆绑模型能够达到与标准模型相当的性能，而自由参数却少得多。因此，假设外显子具有长度为10左右的周期性，看来是有充分依据的。

由于从左到右的构架并不是一个理想的外显子模型，我们希望有一个包含环状构架的模型，这样，对于一个任何给定的外显子，这样的片断可以根据需要被重复多次。参考文献［336］给出了一个用于大肠杆菌DNA的环状构架。图8-16概要显示了一个真正的环状模型的例子。在具体的外显子实验中，环的长度为10，两个旁侧扩展的长度为4。这个模型在训练中采用梯度下降算法，并对主干上的转移进行Dirichlet正则化使其更有利于主状态。由于锚状态在模型连通性中扮演着特殊的角色，需要对它进行附加的正则化。用于锚状态的Dirichlet向量为（0.168 9，0.165 6，0.165 6，0.168 9，0.165 6，0.165 6）。环内部的主状态和插入状态的生成概率分布见表8-3。我们再次获得了与捆绑模型有着显著一致性的结果。从主状态3(M3)开始，模式[^ T] [AT]G清晰可见。

表8-3　环状模型主状态和插入状态的生成分布（见图8-16），训练集为500个长度为100~200个核苷酸的外显子

环状态	A	C	G	T
I1	0.195 7	0.480 8	0.198 6	0.124 9
M1	0.320 7	0.061 5	0.061 9	0.555 9
I2	0.006 2	0.038 1	0.507 9	0.447 8
M2	0.124 6	0.298 2	0.515 0	0.062 2
I3	0.441 2	0.147 4	0.237 7	0.173 7
M3	0.220 8	0.651 9	0.115 9	0.011 4
I4	0.274 3	0.589 3	0.067 6	0.068 9
M4	0.370 9	0.011 3	0.060 3	0.557 5
I5	0.138 9	0.294 6	0.037 8	0.528 7
M5	0.021 9	0.012 1	0.917 9	0.048 1
I6	0.015 3	0.951 9	0.005 2	0.027 7
M6	0.090 5	0.149 2	0.701 7	0.058 6
I7	0.186 2	0.370 3	0.303 7	0.139 9
M7	0.399 2	0.283 5	0.311 9	0.005 5
I8	0.250 0	0.438 1	0.296 8	0.015 1
M8	0.466 5	0.004 3	0.140 0	0.389 1
I9	0.689 2	0.015 6	0.291 2	0.004 0
M9	0.012 1	0.200 0	0.775 9	0.012 0
I10	0.202 8	0.370 1	0.011 7	0.415 5
M10	0.350 3	0.345 9	0.2701	0.078 7
I11	0.144 6	0.685 9	0.086 1	0.083 4

表8-4比较了不同模型训练集负对数概率的累计值随时间变化的情况，实验涉及三个模型：一个自由模型、一个捆绑模型和一个环状模型。虽然正如我们所

预期的——在12个训练周期后，自由模型达到了最优分值，然而这似乎是由某种程度上的过拟合造成的。环状模型的性能在前几个训练周期中优于自由模型。捆绑模型也是如此，只是较环状模型差些。环状模型在所有训练周期中都优于捆绑模型。自由模型只是在第7个训练周期之后才获得比环状模型更好的分值。这也显示出，对于这一数据集，环状模型是一个更好的模型。

表8-4　12次迭代中概率的负对数分值（NLL）的演化情况

周　期	NLL自由模型	NLL捆绑模型	NLL环状模型
1	1.013e+05	1.001e+05	9.993e+04
2	1.008e+05	9.902e+04	9.886e+04
3	9.965e+04	9.884e+04	9.873e+04
4	9.886e+04	9.875e+04	9.859e+04
5	9.868e+04	9.869e+04	9.855e+04
6	9.854e+04	9.865e+04	9.849e+04
7	9.842e+04	9.862e+04	9.848e+04
8	9.830e+04	9.861e+04	9.852e+04
9	9.821e+04	9.860e+04	9.845e+04
10	9.810e+04	9.859e+04	9.842e+04
11	9.803e+04	9.859e+04	9.844e+04
12	9.799e+04	9.859e+04	9.843e+04

η=0.01，使用梯度下降算法，分别为自由模型（见图7-2）、捆绑模型（见图8-15）和环状模型（见图8-16）。所有模型的训练集均为阅读框中500个长度为100~200的外显子。

最后，我们还在外显子和内含子上训练了一种不同类型的环状模型。这个HMM构架为一个给定主状态数量的轮状结构，没有线性排列的旁侧状态，主状态和插入状态间没有区别，也没有删除状态。因而没有与潜在的哑环相关的问题。序列可以从任何一点进入轮状结构。进入点当然可以通过动态规划来决定。通过试验状态数量不同的轮状模型和比较训练集负对数概率的累计数值，可以揭示出最可能的周期性。如果9个状态的轮比10个状态的轮性能更好，我们就可以假设与周期性相关的是三联体阅读框而不是DNA结构方面的特性（见下文）。

图8-17显示了轮状模型的构架（这里的长度为10个核苷酸），其中序列可以从轮上的任何一点进入。来自外部的箭头的线宽代表从相应状态开始的概率。训练完成后，轮状模型的生成参数在外显子模型（上图）的状态8、9和10以及内含子模型（下图）的状态7、8和9，清晰地显示出[^ T] [AT]G的模式。通过训练许

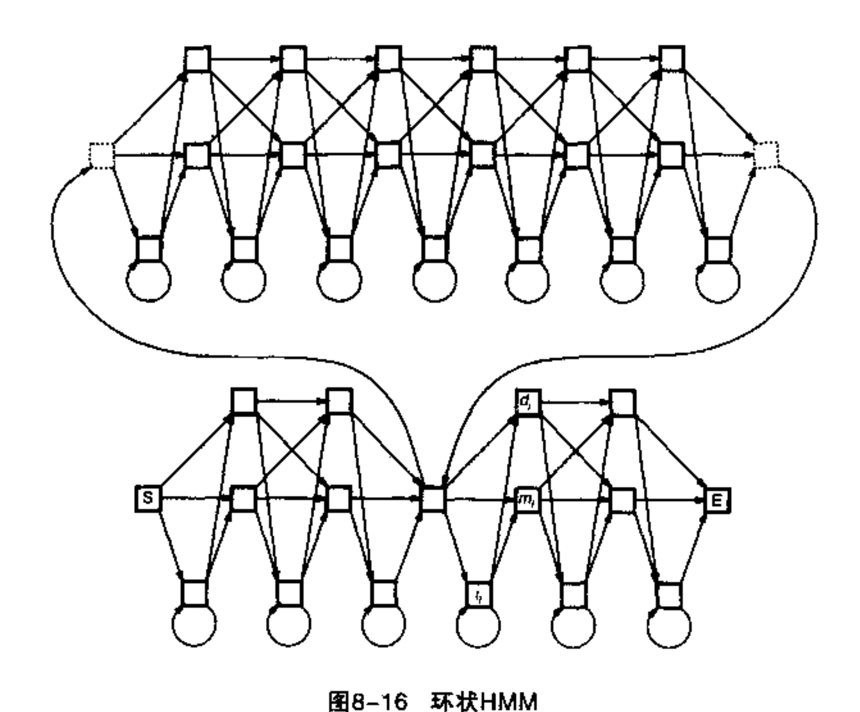


图8-16　环状HMM

由两个旁侧模型和一个环状模型通过一个哑状态连接起来构成，旁侧模型和环状模型都类似于标准构架。

多不同长度的轮状模型，我们发现长度为10的模型对数据适应得最好。跳跃概率（skip probability）在这些模型中都不高的事实，隐含地证实了这一结论。也就是说，如果数据以9为周期，那么长度为10的轮状模型为了能够与数据相适应，在轮中跳过一个状态的可能性会很大。在9个状态的轮中的状态重复与在10个状态的轮中的状态跳跃是不等价的。这些轮状模型都不包含独立的插入状态（如在从左向右的线性HMM中的插入状态）。对同一状态的重复不会给出相同的自由度，这点对可能性而言就好像允许独立的插入状态一样。进而，与传统的多重序列比对中的间隙惩罚（gap penalty）相类似，HMM训练过程也引入一个有利于主状态（惩罚跳跃状态）的正则项。

我们在起始于阅读框中三个密码子位置之一的多个不同外显子的子集上重复上述全部实验，观察到的生成概率模式未见显著变化。为了便于比较，图8-18显示了一个9个状态的轮状模型的生成概率，模型训练集为多联外显子（concatenated exons）的完整mRNA序列的编码部分。这一模型清楚地识别出三联体阅读框（与图6-11相比较）。这一模式出现在内含子序列的事实，进一步否定了上述外显于模式的阅读框相关性起源。

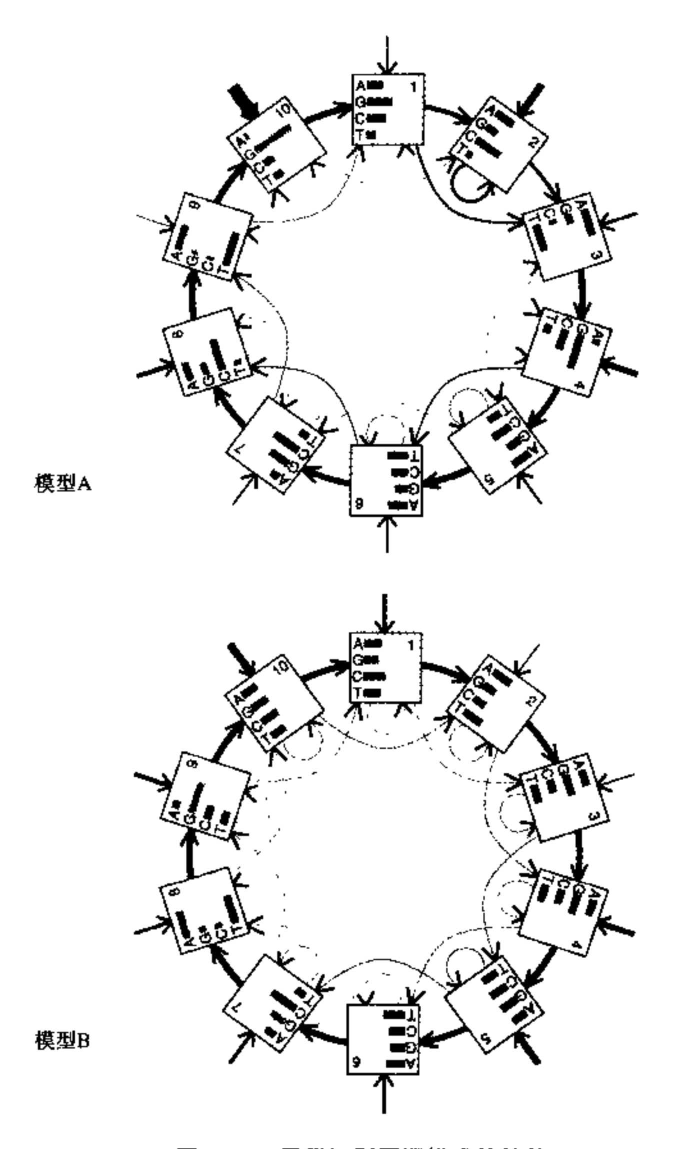


图8-17 用于识别周期模式的轮状HMM

模型A. 周期为10个状态的轮状模型，由训练500个长度为100~200的内部外显子序列获得。不完全比对和阅读框推理导致模式的特性出现在状态2、3和4，以及状态8、9和10。模型B. 周期为10个状态的轮状模型，由训练2 000个人类基因内含子序列获得。在5'端和3'端去除25个核苷酸，以避免剪接位点上的保守序列模式的影响。

实验显示上述周期性在外显子中最强，而且有可能存在于内含子序列中与外显子直接相连的部分，但对于任意选择的深入到内含子内部的片断，这个周期性

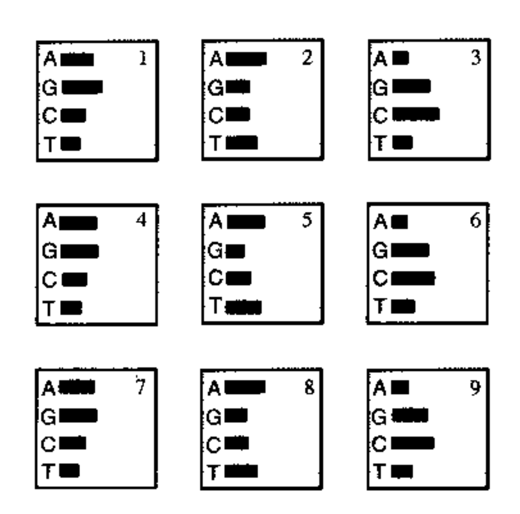


图8-18 包含9个状态的轮状模型的生成分布

模型由训练完整的mRNA序列获得，并且不包含跳跃和自环。周期为3的阅读框模式清晰可见，频率较高的A和G，A和T，以及C和G分别出现在第一、第二和第三密码子位置上。

通常要弱一些。所有使用简单线性从左向右HMM构架的实验，在非编码区都没有发现任何清晰规则的振幅摆动模式。利用轮状模型对不同类型的序列（包括不同类型的外显子、内含子和基因内区域）估计每个核苷酸的概率的负对数的平均值，我们发现周期模式在外显子中最强。比对中的周期（状态9的核苷酸之间的平均距离）显示约为10.1~10.2个核苷酸。

我们都知道“弯曲的DNA”（bent DNA）需要许多相位相同的独立的小弯曲。[488]只有当弯曲的相位周期约为10.5bp（对应于双螺旋中一个完整的螺旋）时，才能得到大范围上稳定的曲率。应用轮状模型对外显子和内含子进行比对，发现序列的周期性有一种潜在的结构性含义，因为比对序列的弯曲相位周期都近似为10，显示出相同的周期性。序列的可弯曲性根据序列相关的三联体可弯曲性参数估计获得，这个参数估计是从DNaseI消化数据（DNaseI digestion data）推断出来的。DNaseI作用于小沟（minor groove）的表面，使DNA分子弯曲并远离酶分子。实验[96]定量地揭示出：对于32种双链三联体，如AAA/ATT、AAA/TTT和CCA/TGG等，可弯曲性参数在某一范围之内变化，数值较低表示缺乏弯曲的潜质，数值较高则对应于指向大沟（major groove）的较大弯曲或可弯曲性。外显子和内含子序列可弯曲性分布图（bendability profile）与核小体定位有关。[34]这些编码区和非编码区中信号强度的差异，对于转录机制识别基因可能有特殊的含义。

8.2.4 人类启动子区的HMM

我们用人类DNA启动子区的序列训练了一些HMM。在一个实验中，启动子数据来自GenBank。[62]所有序列包含转录起始点（由实验确定）的上游和下游至少各250个核苷酸，那些包含非核苷酸字符的序列都被除去了。应用参考文献［259］中的第二种Hobohm算法和参考文献［422］中介绍的一种新颖的发现相似性片断（similarity cutoff）的方法，可以谨慎地降低冗余度。简单地说，这种方法基于对数据集进行完全的逐项比对，用一个极值分布来拟合比对结果的Smith-Waterman分值，[9,550]最后从上述分布中选择一个值使得序列比预期的更为常见。用留下的长度均为501的625个序列，训练一个长度为500的标准线性构架（详见参考文献［421］）。为了使训练更有效，我们用经过实验验证的包含TATA框（TATA-box）的启动子序列的保守序列概率，来初始化TATA框中主状态上的生成分布。

可弯曲性分布图可以通过训练完成的HMM直接计算获得（参见附录D），也通过HMM驱动的多重序列比对获得。图8-19显示了一个由多重序列比对导出的

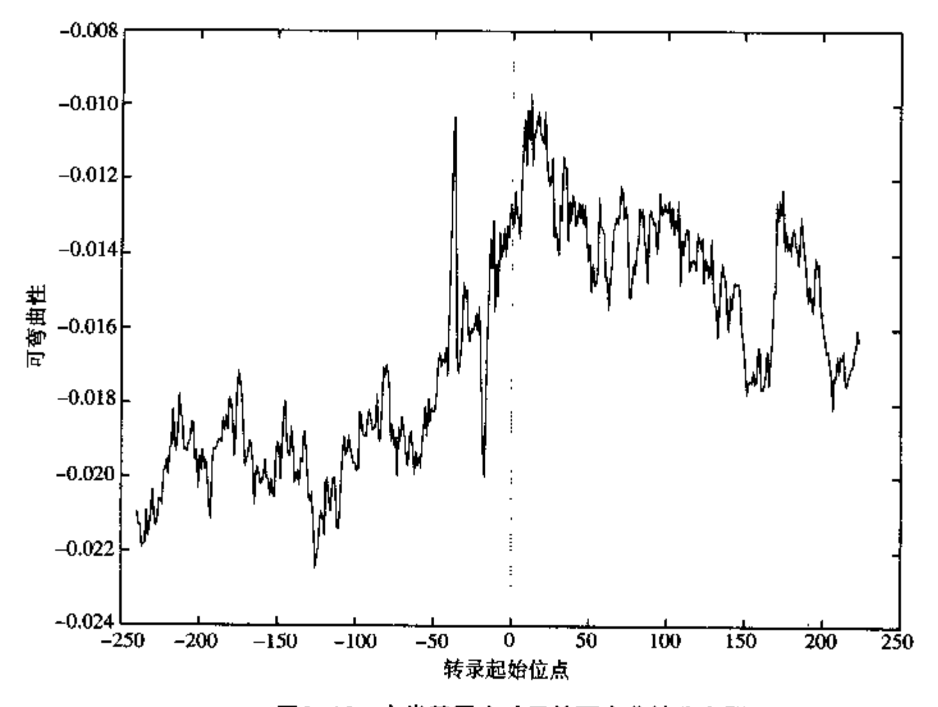


图8-19 人类基因启动子的可弯曲性分布图

转录起始位点大约在图的中央。总体上，可弯曲性在起始位点附近沿下游方向显著增加。这个平均分布图是根据多重序列比对绘制的。根据生成概率而不是实际的三联体频率，计算获得的分布图同样呈现出非常相似的可弯曲性模式。

分布图。图中最引人注目的特性是：在临近转录起始点的下游区域中，可弯曲性显著上升。启动子的特征常常由上游序列中的一些模式和组成倾向性决定。有趣的是，HMM比对的结果预示启动子的下游序列带有某种结构上的相似性，而这些启动子序列原本并不相关。例如，它们并不倾向于与带有某种特定功能的基因相关。通过仔细分析序列的周期性，我们猜测下游区可弯曲性的升高可能与核小体定位以及（或者）与涉及转录启动的其他因素的相互作用有关。我们还用不同的物理标度，如堆积能[410]、核小体定位[218]和螺旋桨式扭曲（propeller twist）[241]等，由HMM主干上的概率计算出一些相似的分布图。所有的分布图一致显示在转录起始点附近有一个强大的信号，以此区分上游和下游。更多的结果，包括周期模式，在参考文献［421］中有更详细讨论。（关于在序列分析问题中如何应用附加的、结构的或其他标度的一般性处理方法，见参考文献［30］。）

8.3　HMM的优势和局限性

8.3.1　优　势

现在，人们已经看到HMM在计算分子生物学中有许多优势。HMM有坚实的统计学基础和有效的学习算法。通过引入局部可学习概率（locally learnable probability），HMM允许以统一的形式对插入和删除进行补偿。模型可以直接从原始数据中学习获得。不同于传统的有监督的神经网络学习，HMM可以兼容不同长度的输入序列，并且不需要一个指导者。它是序列分布图最灵活的一般化推广。HMM可被高效地用于许多任务，从多重比对、数据挖掘和分类到结构分析和模式发现。HMM还能够方便地组合成模型库以及模块或层次结构。

8.3.2　局限性

虽然HMM有很多成功的应用，但它仍受限于以下两方面的弱点：

第一，它们经常含有大量无结构的参数。在蛋白质模型中，图7-2的构架总共包含大约49N个参数（40N个生成参数和9N个转移参数）。对于一个典型的蛋白质家族，N大约为数百，这直接导致模型的自由参数超过10 000。当只能在一个家族中获得很少的序列时（这种情况在基因组计划的早期并不少见），数据不足可能成为一个严重的问题。然而，我们应该看到，一个典型的序列能够提供2N个约束，因此大约25个序列就可以提供与HMM参数数量相当的训练样本。

第二，1阶HMM受其1阶马尔可夫性质的限制，即它们无法表示隐状态之间的依赖关系。蛋白质是通过折叠成复杂的三维形状来决定其功能的。在多肽链上，

可能存在单HMM难以达到的微妙的长程相关性。例如，假设一旦在位置i上发现X，随后通常在位置j上发现Y；而一旦在位置i上发现X′，随后通常在位置j上发现Y′。一个典型的单HMM在位置i和j上有两个固定的生成概率向量。因此无法捕捉到这样的相关性。带有合理约束的HMM仅仅能够表达可能的序列空间上极少的一些分布。[7]然而，我们也注意到，HMM能够轻易地捕捉到一个序列家族中不变的长程相关性，即便这一相关性是由三维相互作用导致的。例如，对于一个蛋白质家族中的两个线性间隔的区域，由于在三维结构中彼此靠近，它们一定有共同的亲水性模式。相同的亲水性模式将在家族的所有成员中出现，并且很可能反映在相应HMM训练后的生成参数上。

第9~11章尝试着超越HMM，具体的方法包括：将HMM与神经网络组合以构造混合模型（第9章），对进化过程建模（第10章），以及扩大HMM的生成规则集合（第11章）。

⑦ 任何分布都可以用一个指数规模的HMM表示。其中，一个初始状态与不同的确定性状态序列相连接；对于任意可能的符号序列，有一个与序列本身的概率相等的转移概率。

第9章 生物信息学中的概率图模型

9.1 生物信息学中的图模型概述

把贝叶斯系统应用到典型的现实问题时，我们所遇到的首要障碍之一是高维概率分布问题。这是因为我们所获得的数据是高维的，所用的模型也是高维的，通常问题所涉及的参数有几千个甚至更多。高维也来自于其他一些被称为隐变量的变量。一般地，高维造成的全局分布$\mathbf{P}(D, M, H)$在数学上是很难处理的，而这正是图模型理论发挥作用的地方。根据在现实世界中大多数依赖关系是局部依赖关系这样一个事实，高维分布可以用定义在较小空间上一簇变量分布的乘积进行估计，这使问题容易处理。[348,292] 例如，在标准马尔可夫模型中，$t+1$时刻的现象仅仅通过现在t时刻的现象与过去发生联系。因此，全局概率分布$\mathbf{P}(X_1, \cdots, X_N)$可以分解成形式为$\mathbf{P}(X_{t+1}|X_t)$的局部概率分布的乘积。

为了更明确起见，我们把注意力集中在一类特殊的图模型，即贝叶斯网络（图模型的更正式的处理见附录C）。[416] 贝叶斯网络由带有N个节点的有向无环图组成。每一个节点联系着一个随机变量X_i。模型的参数是每一个随机变量的局部条件概率或特征，这些随机变量由与父节点$\mathbf{P}(X_i \mid X_j\text{: } j \in N^-(i))$相关联的随机变量给出，其中$N^-(i)$表示节点$i$的所有父节点的集合。贝叶斯网络的马尔可夫独立性假设等价于以下的全局因子分解特性：

$$\mathbf{P}(X_1, \cdots, X_N)=\prod_i \mathbf{P}(X_i|X_j\text{: } j \in N^-(i)) \tag{9.1}$$

换言之，全局概率分布是所有局部特征的乘积。实际应用上，在贝叶斯网络中，

边的方向用于表示因果关系或时间延续关系。因此，贝叶斯网络被大量用于建立生物序列模型的时代已经到来，这一点并不令人感到惊奇。此外，贝叶斯网络还以类似的方式用于建立语音模型或其他序列相关模型，以及用于构造专家系统。

事实上，对于生物（或其他）序列，贝叶斯体系允许我们为之构建一组复杂度递增的贝叶斯网络模型。这组模型的等级结构基于这样的事实：在某些层次上，生物序列具有顺序排列的主体结构。我们考虑的关于生物序列的最简单的概率模型，是第3章中涉及的具有4个面（代表DNA的4种核苷酸）或20个面（代表蛋白质的20种氨基酸）的骰子模型（见图3-1）。这种模型可以表示为单个节点或多个不连通的相同节点组成的贝叶斯网络（后者效果更好），每个节点对应于序列或序列族中的一个位置。尽管骰子模型非常简单而且远离实际的生物序列，但它是研究生物序列问题的第一步，并且通常被当做背景模型与更复杂的方法进行比较。

更进一步，我们可以设想一个由不同骰子组成的序列，每一骰子代表序列中的一个位置。这基本上就是我们用于生成序列谱的模型，例如对已有的多重序列比对进行抽象。如果我们将模型中的节点连接成一条从左到右的链，就得到了标准的1阶马尔可夫模型。建立2阶和更高阶的马尔可夫模型也是可能的，在这类模型中，现在状态可能依赖于直接相邻的前几个过去状态。这类模型的贝叶斯网络表示的主要缺点显而易见，即随着阶数的增长，模型的参数空间呈现组合爆炸。然而，对于像DNA这样字符集比较小的问题，阶数高达6阶的马尔可夫模型仍然是可行的，并且常见于有关基因发现等应用的文献中（见图9-1）。

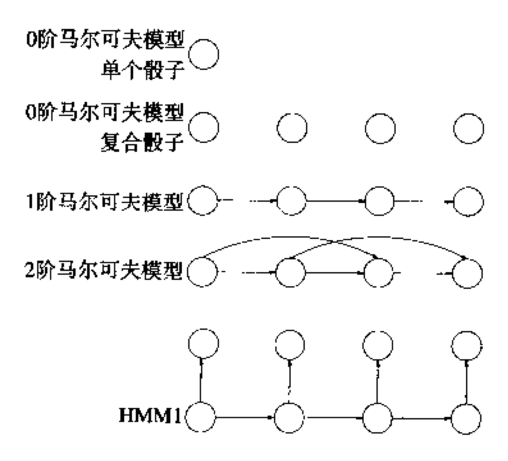


图9-1 复杂度递增的马尔可夫模型的贝叶斯网络表示

阶数为0的马尔可夫模型对应于单个骰子或独立骰子的集合。阶数为1的马尔可夫模型对应于标准的1阶马尔可夫链。在阶数为2的马尔可夫模型中，现在状态依赖于前2个过去状态。所有阶数为1的HMM均具有这里给出的相同的贝叶斯网络表示。

然而，简单的从左到右的马尔可夫模型并不直接反映插入和删除操作。我们已经看到，这类操作可以通过隐马氏模型（HMM）进行处理。HMM很容易表示为贝叶斯网络。贝叶斯网络对其他模型的表示也与之类似，如Kalman滤波器。HMM的贝叶斯网络表示阐明了它们的概率结构、相应信息的传播以及学习算法，例如著名的前向—后向算法和EM/梯度下降法的其他各种变体。[493]

更复杂的马尔可夫模型已经被用于人工智能，例如输出依赖于两条或多条前向马尔可夫链的因子HMM。例如，在语音相关的领域中，可以用一条链表示音频信息，用另一条链表示口形的视频信息。[203,205] 参考文献［40,58］中描述了另一类称为输入-输出HMM（IOHMM）的模型，我们将在后续章节中讨论这类模型（见图9-2）。这类模型能将一个给定的输入序列翻译成一个能定义于不同字符集上的输出序列。

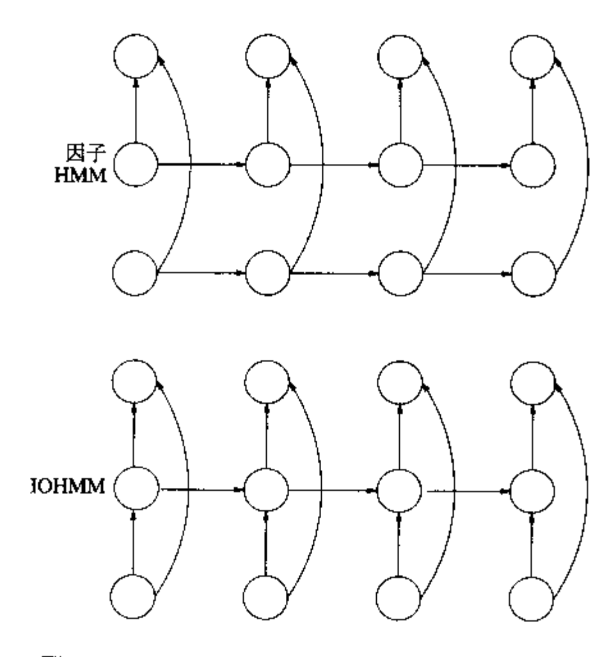


图9-2　因子HMM和IOHMM的贝叶斯网络表示

关于生物序列的一项重要观察结果指出，实际的生物序列基于某种空间结构，而不是时间结构。特别是来自“将来”的信息可以用来解释“现在”的现象，而不打破任何因果性的约束。根据这一点，至少可以在前面的模型中引人后向马尔可夫链。但在引人的过程中必须非常小心，因为很容易证明：通过变量替换，一条简单的后向马尔可夫链可以完全等价于一条对应的前向马尔可夫链。而这两个相应的贝

叶斯网络模型的参数可以通过贝叶斯定理相联系。同样地，如果改变一个因子HMM中一条链的方向，我们将得到另一个完全相同的因子HMM，而实际上我们几乎什么也没得到（见图9-3）。然而，如果在一个IOHMM中引入一条后向链，我们将获得一类新模型，称为双向输入-输出HMM（BIOHMM）（见图9-4）。[36]

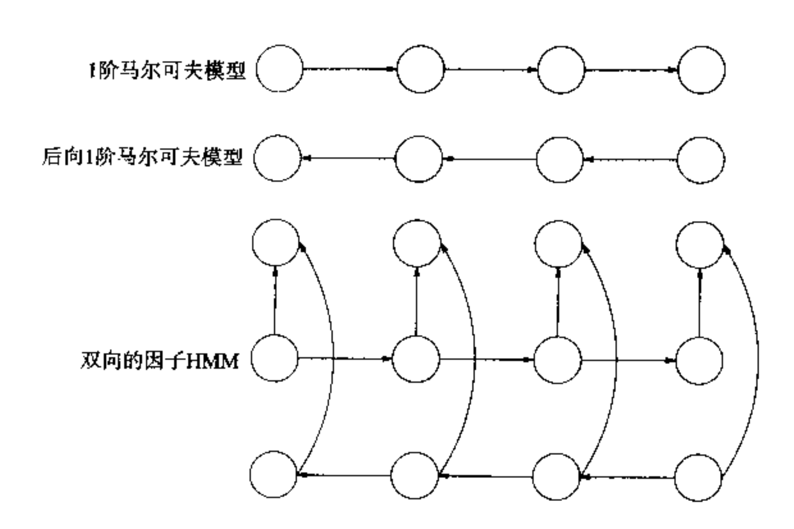


图9-3 后向马尔可夫链的贝叶斯网络表示

图中的所有后向链都可以通过简单的变换由前向链替代。

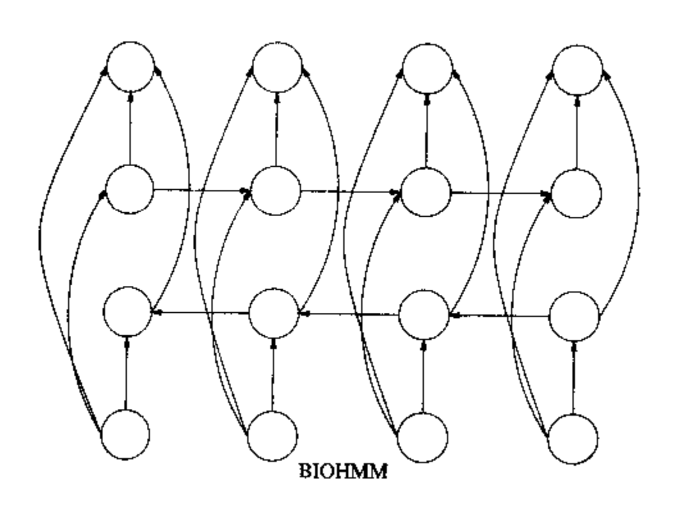


图9-4 一个BIOHMM的贝叶斯网络表示

注意图中存在大量的无向环。

我们将在本章的最后一节中讨论BIOHMM及其相关的模型在蛋白质二级结构预测中的应用。下面我们将首先介绍概率图模型在序列分析中的应用，特别是DNA的对称性、基因发现、基因分析以及结合神经网络与图模型的一般技术。

9.2 马尔可夫模型与DNA的对称性

在一段双螺旋的DNA链中，碱基A的数目与碱基T的数目相等，碱基C的数目与碱基G的数目相等。在20世纪50年代早期，正是这个今天看起来微不足道的性质指导沃森和克里克发现了DNA的双螺旋结构。这一性质也以夏加夫第一奇偶校验准则（Chargaff's first parity rule）的形式为人们所熟知。[119] 然而，夏加夫第二奇偶校验准则（Chargaff's second parity rule）却很少为人所知。后者指出，同样的规律在长度合适的DNA单链中也近似成立。这一准则首先在20世纪60年代提出，[303,120] 近年来已经得到了一定程度的认可。[430,185,231]

夏加夫第二奇偶校验准则的正确性可以通过研究不同的组织、DNA的不同类型（如编码和非编码DNA）以及不同长度范围的DNA等来确定。为简单起见，这里我们只考察酵母基因组中的DNA。如果分别测量酵母的每条染色体的沃森和克里克链中DNA的组成成分，我们将发现这些组成成分相当稳定，而且符合夏加夫第二奇偶校验准则，即A和T各占大约30%，C和G各占大约20%（表9-1）。同样的对称性在酵母线粒体DNA中也被观察到，但是组成的比例有所差别。同样地，在其他组织的单链DNA中，其组成成分有所不同但仍具有对称性。

为了研究双链DNA的对称性，我们统计某一给定长度DNA的每条链上碱基出现的频率。这些频率对应于1阶马尔可夫模型。当然，我们也可以通过考察对应于N个连续状态的统计结果来研究高阶（阶数为N）马尔可夫模型。特别地，我们可以考察，除1阶马尔可夫模型之外，夏加夫第二奇偶校验准则是否对高阶也成立，例如考察等价于2阶马尔可夫模型的二核苷酸。

一个阶数为N的DNA马尔可夫模型有4^N个参数与转移概率$\mathbf{P}(X_N|X_1,\cdots,X_{N-1})$相关，也可以写为$\mathbf{P}(X_1,\cdots,X_{N-1}\rightarrow X_N)$。其中，对于字符集中的所有可能的符号$X_1,\cdots,X_N$，有形如$\pi(X_1,\cdots,X_{N-1})$的初始概率分布。由于模型的参数以指数级增长，因此有限的数据集只能确定阶数在一定范围内的模型。例如5阶的DNA马尔可夫模型共有1 024个参数，而10阶的DNA马尔可夫模型的参数则超过100万。相应地，阶数越高，确定合适的模型所需要的数据量就越大。

表9-1 酵母基因组和线粒体中每条染色体的DNA的1阶分布

	A	C	G	T	碱基总数（bp）
1号染色体	69 830	44 641	45 763	69 969	230 203
	30.33%	19.39%	19.88%	30.39%	
2号染色体	249 640	157 415	154 385	251 700	813 140
	30.70%	19.36%	18.99%	30.95%	
3号染色体	98 210	62 129	59 441	95 559	315 339
	31.14%	19.70%	18.85%	30.30%	
4号染色体	476 752	289 343	291 354	474 480	1 531 929
	31.12%	18.89%	19.02%	30.97%	
5号染色体	176 531	109 828	112 314	178 197	576 870
	30.60%	19.04%	19.47%	30.89%	
6号染色体	82 928	52 201	52 435	82 584	270 148
	30.70%	19.32%	19.41%	30.57%	
7号染色体	338 319	207 764	207 450	337 403	1 090 936
	31.01%	19.04%	19.02%	30.93%	
8号染色体	174 022	109 094	107 486	172 036	562 638
	30.93%	19.39%	19.10%	30.58%	
9号染色体	134 340	85 461	85 661	134 423	439 885
	30.54%	19.43%	19.47%	30.56%	
10号染色体	231 097	142 211	143 803	228 329	745 440
	31.00%	19.08%	19.29%	30.63%	
11号染色体	206 055	127 713	126 005	206 672	666 445
	30.92%	19.16%	18.91%	31.01%	
12号染色体	330 586	207 777	207 064	332 745	1 078 172
	30.66%	19.27%	19.21%	30.86%	
13号染色体	286 296	176 735	176 433	284 966	924 430
	30.97%	19.12%	19.09%	30.83%	
14号染色体	241 561	151 651	151 388	239 728	784 328
	30.80%	19.34%	19.30%	30.56%	
15号染色体	339 396	209 022	207 416	335 449	1 091 283
	31.10%	19.15%	19.01%	30.74%	
16号染色体	293 947	180 364	180 507	293 243	948 061
	31.01%	19.02%	19.04%	30.93%	
线粒体染色体	36 169	6 863	7 813	34 934	85 779
	42.17%	8.00%	9.11%	40.73%	
16条核染色体	3 729 510	2 313 349	2 308 905	3 717 483	12 069 247
	30.90%	19.17%	19.13%	30.80%	
所有的染色体	3 765 679	2 320 212	2 316 718	3 752 417	12 155 026
	30.98%	19.09%	19.06%	30.87%	

由于两条链的互补性，一条链上阶数为N的马尔可夫模型直接定义了对应的反向互补链上阶数为N的马尔可夫模型。如果一条链上的N阶模型与其反向互补链上的N阶模型完全一致，我们则称该模型是对称的。事实上，当且仅当$\mathbf{P}(X_1,\cdots,X_N)=P(\overline{X_N},\cdots,\overline{X_1})$时，马尔可夫模型才是对称的。例如考察酵母基因组的DNA序列，在阶数不小于9的所有高阶马尔可夫模型中，我们发现很高程度的对称性，即便在各种子序列中也是如此。（表9-2）有人认为这种对称性可以通过1阶对称性来解释。事实上，如果$\mathbf{P}(A)=\mathbf{P}(T)$和$\mathbf{P}(AA)=\mathbf{P}(A)\mathbf{P}(A)$，则有$\mathbf{P}(AA)=\mathbf{P}(TT)$。这个问题的准确提法是：高阶马尔可夫模型是否可分解，即高阶马尔可夫模型能否通过低阶模型的乘积完全决定。

表9-2　2阶转移参数和酵母上游区500 bp的二核苷酸分布

A→A	0.364 3	AA	0.115 4
A→T	0.280 6	AT	0.088 9
A→G	0.185 8	AG	0.058 9
A→C	0.168 4	AC	0.053 3
T→A	0.260 2	TA	0.081 4
T→T	0.366 2	TT	0.114 6
T→G	0.185 8	TG	0.058 1
T→C	0.188 2	TC	0.058 9
G→A	0.316 6	GA	0.058 1
G→T	0.278 4	GT	0.051 1
G→G	0.194 5	GG	0.035 7
G→C	0.210 6	GC	0.038 7
C→A	0.330 4	CA	0.061 9
C→T	0.311 6	CT	0.058 3
C→G	0.163 9	CG	0.030 7
C→C	0.194 1	CC	0.036 4

更正式地，一个阶数为N的马尔可夫模型导出低阶长度为M的词（M-mers）的分布，称为原始分布的限制或投影。这种投影很容易得到，例如可以利用阶数为N的马尔可夫模型生成一个长字符串并计算长度为M的词的统计值。特别地，从阶数为N的马尔可夫模型导出的1阶平衡分布一定满足以下平衡方程：

$$P(X_2,\cdots,X_N)=\sum_Y P(X_N|Y,X_2,\cdots,X_{N-1})P(Y,X_2,\cdots,X_{N-1}) \qquad (9.2)$$

如果N阶马尔可夫模型是对称的，则它的低阶限制或投影也是对称的。然而，反之却不成立。一般地，一个阶数为N的对称马尔可夫模型能够以多种形式扩展为M阶马尔可夫模型，其中$M>N$，但扩展后的模型不一定是对称的。因此，酵母的1阶分布具有对称性，并不意味着它的2阶分布也是对称的。然而，一个给定的

阶数为N的马尔可夫模型有惟一的到阶数为M（$M>N$）的马尔可夫模型的因子扩展。例如，一个由参数p_X（p_A，p_C，p_G，p_T）定义的1阶马尔可夫模型有一个带有参数$p_{XY}=p_Xp_Y$的2阶因子扩展。

对于一个给定的阶数为N的马尔可夫模型，通过任意低阶的马尔可夫模型，我们都可以提取它的对称因子。对于每一个长度为N的词（N-mers）和它的反向互补链，我们可以得到由N阶马尔可夫模型确定的期望次数与由被用做因子分解的M阶马尔可夫模型确定的期望次数之间的比值。对称性的残差可以通过长度为N的词及其反向互补链之间这一比值的相关性来度量。如果对酵母应用这种方法，我们发现在高阶模型中存在大量对称性的残差，这点不能通过1阶成分的对称性等因素加以解释（表9-3）。

表9-3　次数与对称现象

	2	3	4	5	6	7	8	9
0	1.0	.99	.99	.99	.99	.99	.97	.95
1	.98	.97	.97	.97	.95	.90	.77	.55
2		.94	.95	.94	.91	.83	.66	.45
3			.97	.94	.89	.77	.57	.36
4				.82	.73	.58	.39	.24
5					.60	.46	.29	.18
6						.34	.21	.14
7							.12	.10
8								.09

0行表示邻接的上游链与其反向互补链两者的长度为N的词的次数$C(N=2,\cdots,9)$之间的相关系数。在行$M=1$到8，相似的相关系数用比率$C/E(C)$计算得到，其中$E(C)$是由适合上游区的M阶马尔可夫模型产生的次数(C)的期望值。水平方向=词的长度，垂直方向=模型阶数。

因此，高阶马尔可夫模型可以使我们更详细地研究夏加夫第二奇偶校验准则。当然，夏加夫第二奇偶校验准则对于局部序列是不成立的，某些病毒基因组也不满足这一准则。虽然我们都知道，在原核基因组中，复制起始点附近存在成分偏差，但大体上夏加夫第二奇偶校验准则还是显著有效的，这或许是不同尺度下的各种因素综合作用的结果。根据夏加夫第一奇偶校验准则，任何对DNA两条链不加区别的外加作用都将对夏加夫第二奇偶校验准则有所贡献。由辐射导致的突变可能就属于这类情况。同样的，为了产生相同数目的互补碱基对，细胞的复制机制就必须被优化，这也应该支持夏加夫第二奇偶校验准则的1阶形式。人们正在研究这一准则在更大范围内的影响，例如在DNA每一条链上基因的近似对称分布（表9-4），这种分布也能通过概率马尔可夫模型来建模。

表9-4　酵母的每条链和每条染色体中长度大于100的ORF的数目

DNA	W ORF	C ORF	总 数
1号染色体	56	51	107
2号染色体	200	226	426
3号染色体	75	99	174
4号染色体	400	419	819
5号染色体	146	141	287
6号染色体	67	67	134
7号染色体	298	273	571
8号染色体	153	131	284
9号染色体	106	118	224
10号染色体	201	186	387
11号染色体	175	161	336
12号染色体	261	286	547
13号染色体	246	244	490
14号染色体	219	201	420
15号染色体	295	278	573
16号染色体	256	244	500
总 数	3 154	3 125	6 279

统计中排除了tRNA和rRNA基因，总数中不包括线粒体染色体。

9.3　马尔可夫模型和基因发现程序

马尔可夫模型和图模型在基因分析中最重要的一类应用是构造各种基因发现和基因分析程序，已有的程序包括GeneMark和GeneMark.hmm[81,82,367]、GLIMMER[461]、GRAIL[529]、GenScan[107]和现在的GenomeScan、Genie[441]等。本章中，我们的目的不是给出所有基因发现程序详尽的列表或详细描述这些程序，也不在于比较它们各自的优缺点。我们旨在提供一个全面的概述，以展示如何通过概率图模型来构造和理解各种基因发现程序。

集成的基因发现和基因分析程序一般具有模块结构，而且通常采用相同的基本设计策略。它们包括两类基本模块，分别用于发现边界元素和可变长度区域。与局部信号相关的边界模块包括：剪接位点，起始和终止密码子，各种转录因子和其他蛋白质结合位点（例如TATA框），转录起始位点，分支点，转录终止子，聚腺苷酸化位点（polyadenylation），核糖体结合位点，拓扑异构酶Ⅰ剪切位点，拓扑异构酶Ⅱ结合位点等。区域模块通常与外显子、内含子和基因间区域相关。根据众所周知的统计上的差别，外显子模型通常依次分为初始、中间和末端外显

子。最后，整个基因组的计算模型还包括其他一些区域，诸如Alu序列之类的各种重复区域。

图9-5、9-6、9-7和9-8分别给出了各种基因发现程序的高层结构图示。（使

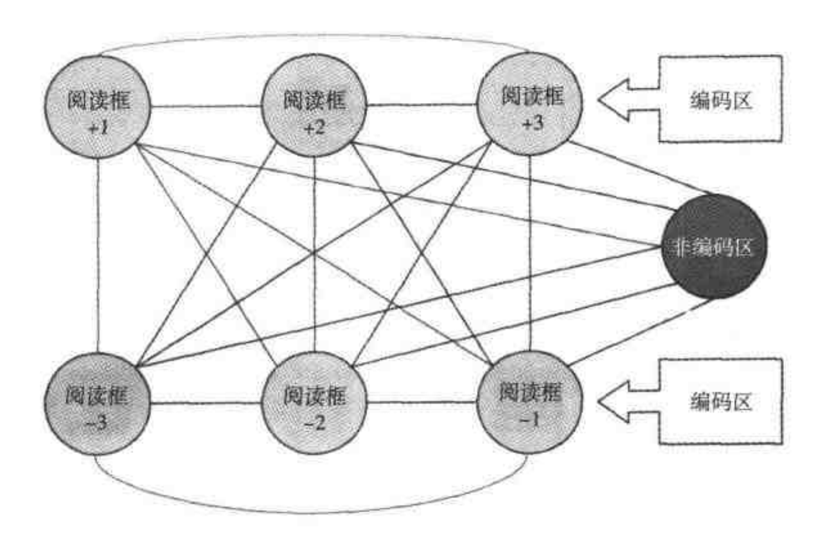


图9-5 原核基因组的GeneMark的图表示

对于原核基因组，典型高层模块包括编码区模块和非编码区模块。

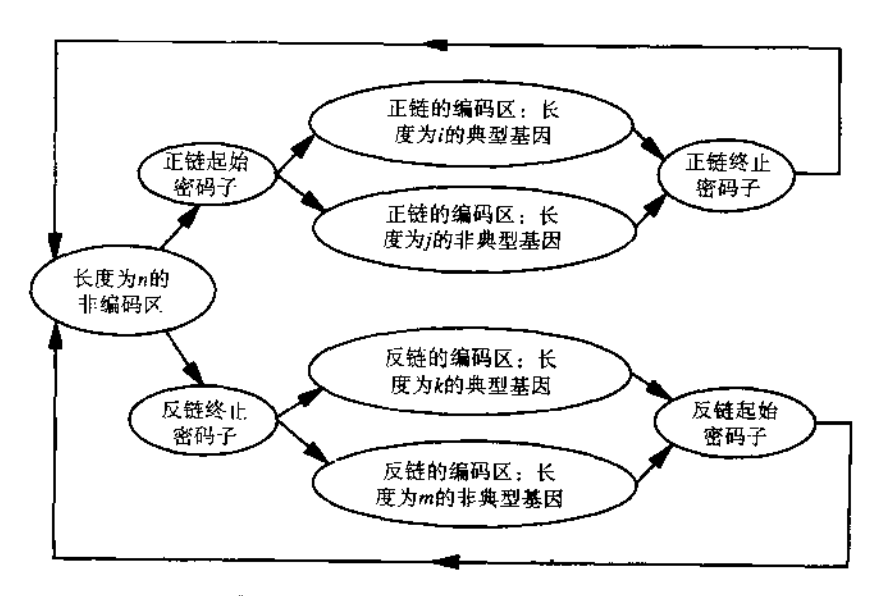


图9-6 原核基因组的GeneMark.hmm的图表示

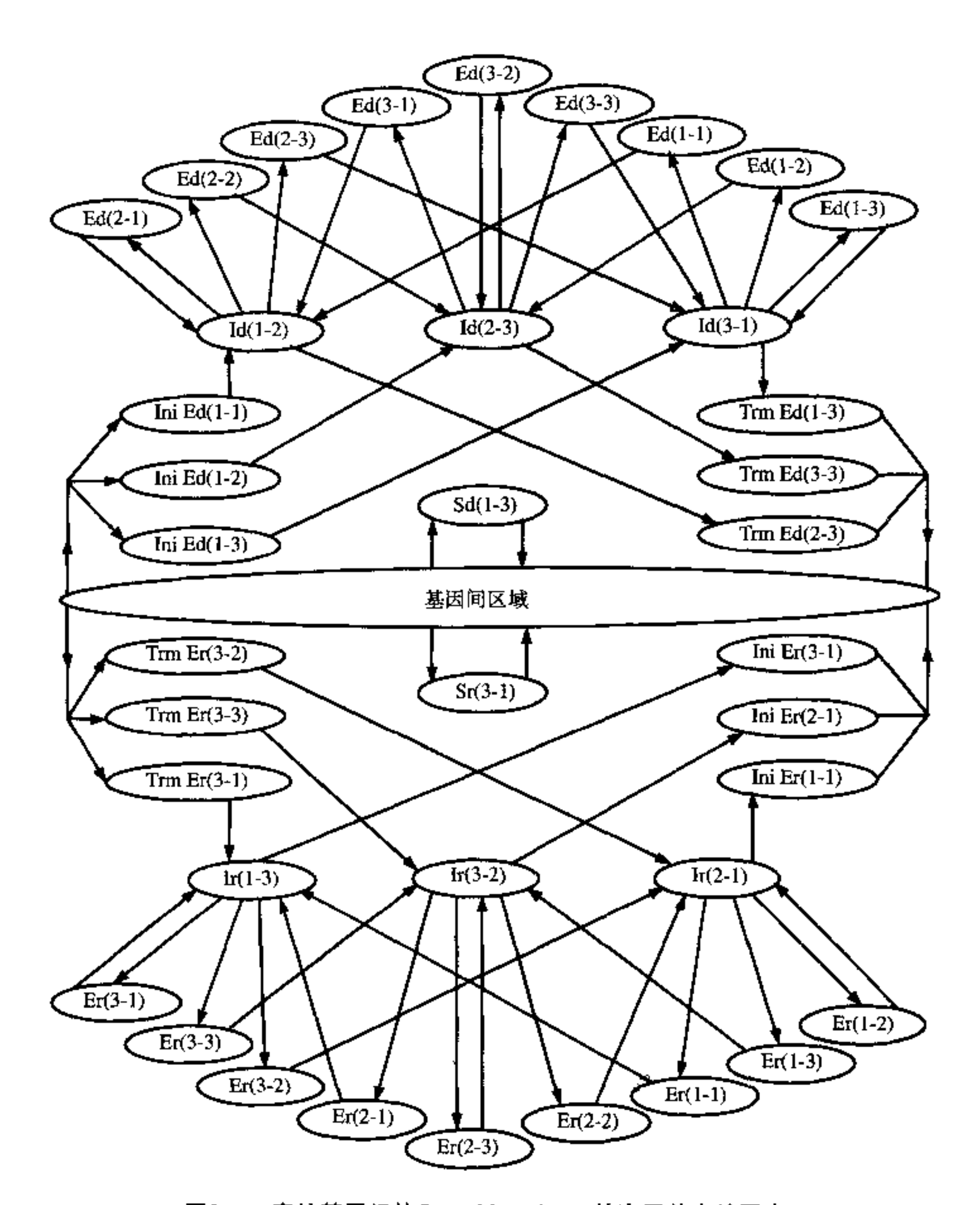


图9–7　真核基因组的GeneMark.hmm的高层状态的图表示

模型包括的状态对应于正向和反向链中所有阅读框中的初始和末端外显子、内部外显子以及内含子。

用这些图表得到了原作者的许可。）由于出现外显子和内含子等原因，真核生物的基因发现程序的高层结构图示及其相关的具体图模型比原核生物要复杂得多。这些示意图中的模型没有直接表示为贝叶斯网络，而仅仅是一种状态转换图，正

如第7章中我们没有用贝叶斯网络表示HMM的标准构架一样，这种表示法我们在本章的第一部分中已经见到了。事实上，至少在某种程度上，大多数基因发现程序可以看做是HMM或HMM的变形。

在这些图示中，高层次的节点代表边界和区域模块。在基因发现程序中，模块的选择和训练实现之间有一些不同。在边界模块情况下，早期的实现用简单的保守序列。后来发展为用序列谱或权重矩阵方法和马尔可夫模型，序列谱或权重矩阵方法是1阶马尔可夫模型的特殊情形，其中的分值采用对数似然度或对数似然度比形式。由于DNA的字符集中只有4种字符，因此当存在足够的训练数据的时候，边界模块的实现也可以用高阶马尔可夫模型。在代数学上，神经网络可以看做是权重矩阵方法的推广，它也可以用在一些边界模块的实现上。

变长区域建模通常用阶数不高于6的马尔可夫模型。特别地，编码区有明显的3和6周期性，这些周期性很容易被组合到3阶或6阶的马尔可夫模型中。外显子模型必须考虑阅读框，阅读框的信息必须以某种方式通过插入序列传递给下一个外显子。状态连续性不但可用于建立不同的阅读框，而且可用于建立每种成分的长度分布模型，即系统在每种状态下应该持续多久。这种持续性也可以通过从已有数据中提取经验分布或者用理论分布拟合训练数据进行建模和调整（见参考文献［154］）。因为基因可以出现在两条链中任意一条从5'到3'的方向上，基因发现程序必须能够以镜像方式在两种情况下建模。一个基因投影到相对应的另一条链上，因此能通过探测一条链找到另一条链上的基因。

利用动态规划和Viterbi寻径法（Viterbi paths）（如最大似然估计、极大后验概率法，甚至在参考文献［339］中叙述的条件极大似然法），这些模型能用于探测和分析大的基因组区。根据这些区域大小的不同，这些方法在计算上可能要求比较高。利用包含在大型EST和蛋白质数据库中关于编码区的信息，诸如Pfam数据库等的HMM模型的数据库以及比对方法，可以进一步筛选和提高性能。各种边界和区域模型的参数选择可以适合不同的组织，甚至适合带有不同成分或不同基因类的基因组区域，最终形成了各种专门的基因发现程序和基因分析程序。

虽然基因发现程序的性能不易度量和比较，但总体上说，基因发现程序的性能比过去几年有了显著提高。现在这些程序在基因组注释计划中起到了重要的作用。尽管如此，仍有一些意义重大的挑战留待我们解决，例如更好地建立调控区模型和选择性剪接模型。

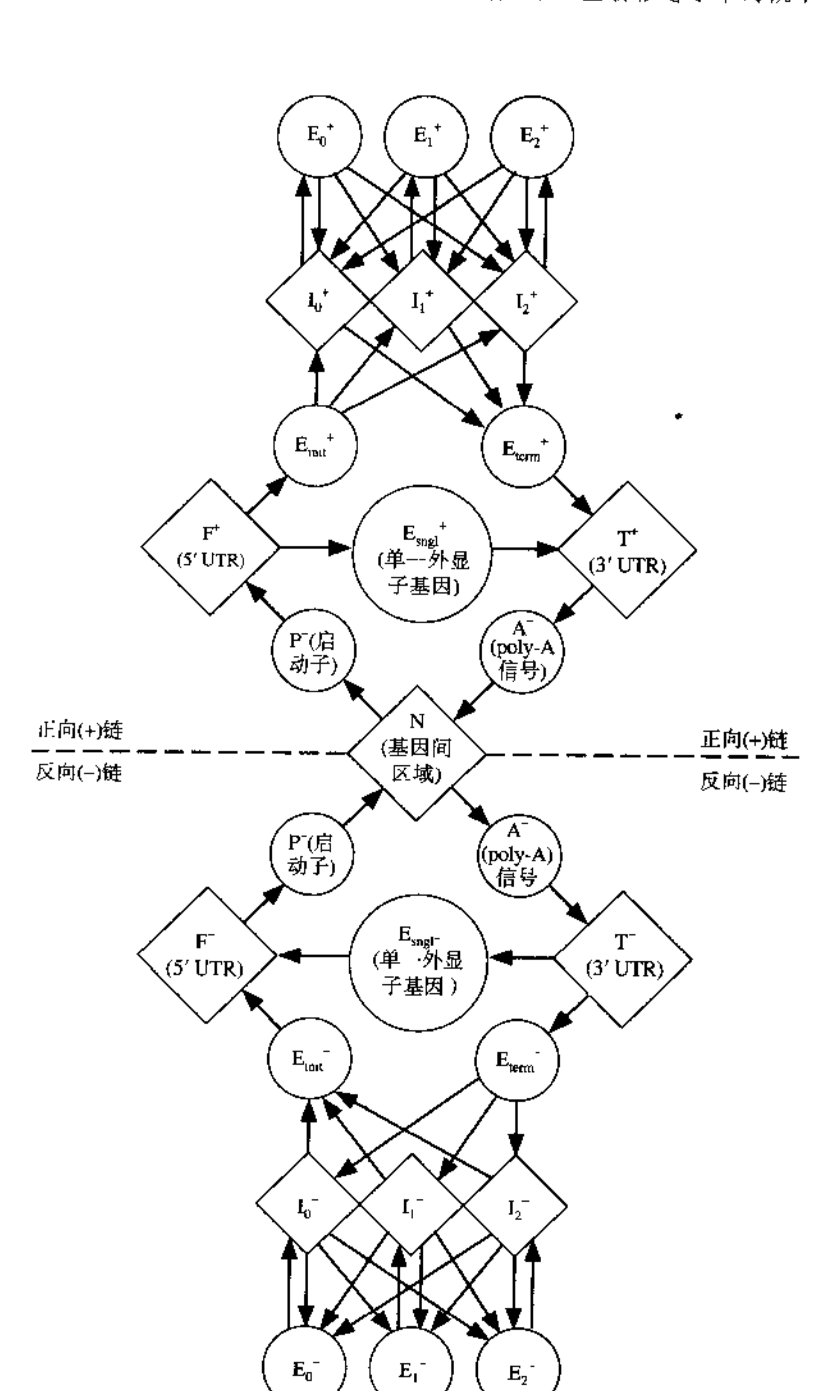


图9-8　GenScan中隐状态的图表示

该图与图9-7类似，注意增加的状态，例如poly-A信号。

9.4 混合模型和图模型的神经网络参数化

9.4.1 一般体系

为了克服HMM的局限性，这里我们考察把HMM和神经网络结合起来形成混合模型的可能性，这种混合模型包含了神经网络的表达能力和HMM处理连续时间序列的能力。在这一部分中，我们大多沿用参考文献［40］中的推导。HMM和神经网络的结合有许多方式。混合模型已经被用于语音和笔迹的识别。[82,126] 在许多这些应用中，神经网络被当做前端处理器提取各种特征，如笔画、字符、音素等。这样，HMM可用在词和语言建模等更高级的处理阶段。⑧虽然有一些例外情况[57]，但HMM和神经网络的组成元件通常分开训练。在参考文献［126］中描述的一种混合模型的不同类型中，神经网络元件用于对由不同HMM产生的似然模式的分类。相反地，这里我们将讨论混合模型中HMM和神经网络元件不可分的情况。在这些结构中，神经网络元件用于重新确定参数和调整HMM元件。两种元件用统一的算法训练，这种统一算法将HMM的动态规划算法和神经网络反向传播算法结合在一起。在我们详细讨论混合模型的细节之前，有必要用第2章及图模型中叙述的一般概率观点来浏览一下这种混合处理方法。

一般的混合体系

从第2章中知道，我们感兴趣的基本目标是数据的概率模型$M(\theta)$，θ为模型参数。然而当模型的复杂性和数据之间不匹配时，问题就出现了。过度复杂的模型导致过拟合，过度简单的模型则导致欠拟合。

一般的混合建模方法试图同时解决这两个问题。当模型太复杂时，利用更简单的参数向量w的函数$\theta=f(w)$，对它重新参数化。这是单模型的情形。当数据太复杂时，由于无法使用其他模型类，解决此问题仅有的方法是以多个$M(\theta)$模拟数据，当$M(\theta)$覆盖数据空间的不同区域时，θ离散或连续地变化。因此，参数必须通过输入函数以及上下文以$\theta=f(I)$的形式进行调整。这是多模型的情形。在一般情形下，两者也许都是可取的，于是有$\theta=f(w, I)$。就函数f可以属于不同的模型类而言，这种处理方法是混合方法。由于神经网络的通用近似性质（见第5章），一种自然的处理方法是用神经网络计算f，但是其他的表示方法可能也适用。由于这使模型重新参数化容易在各个层次进行，因此这种处理方法是分层的。为简单起见，这里我们只限于讨论单层重新参数化。

⑧ 在分子生物学的应用中，神经网络可以令人信服地用于解释各种测序机器的连续输出，但这不是我们在这里关注的问题。

9.5 单模型情形

基本思想

在一般的HMM中，一个生成或转移向量θ仅仅是状态i的函数：$\theta=f(i)$。第一个基本思想是：为了计算HMM的参数，即计算函数f，在HMM的上端加入一个神经网络。神经网络是通用逼近器，因此可以表示任意的f。更重要的一点或许是，参数的神经网络表示可以灵活引入许多可能约束。为了简单起见，我们仅仅在蛋白质序列分析问题中讨论生成参数，但是这种处理方法也可以直接扩展到讨论转移参数和所有其他字符集中。

在对（7.33）的重新参数化中，我们可以考虑每一个HMM生成参数由一个小的神经网络计算得到，这种小的神经网络的一个输入赋值为1（偏倚），没有隐层，有20个softmax型输出节点（图9-9A）。输入与输出之间的连接赋以参数w_{iX}。这些都可以通过任何用于计算HMM参数的复杂神经网络得到直接推广。联系不同

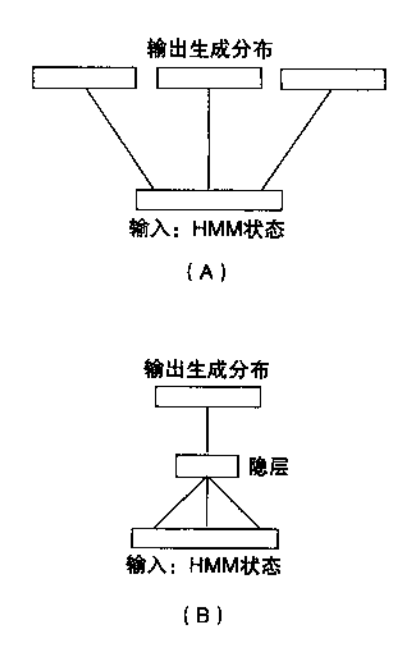


图9-9 从HMM到混合HMM/NN

（A）参考文献［41］中使用的简单HMM/NN混合结构。每一个HMM状态都有自己的NN。这里的NN非常简单，它没有隐层，只有1个softmax型节点组成的输出层，用于计算状态生成或转移参数。为简单起见，该图仅仅表示了生成参数的输出。（B）一个HMM/NN结构的图示，其中和不同状态（或不同状态群）有关的NN通过1个或几个隐层相连接。

状态的神经网络也可以通过1个或多个共同的隐层相连接，而全面的网络构造要根据具体问题而定（图9-9B）。然而，在离散字符集的情形下，例如对于蛋白质的处理，每一状态的生成分布是多项分布，因此对应的网络输出应该由$|A|$个归一化指数型节点组成。

例　子

作为一个简单的例子，考虑图9-10的HMM/NN的混合构造，这种构造由以下部分组成：

1. 输入层：每个节点对应每个状态i。在每一时刻，除了一个赋值为1的节点外，所有节点赋值都为0。如果节点i赋值为1，则网络计算e_{iX}，即状态i的生成分布。
2. 隐层：$|H|$个下标为h的隐节点，每一个节点的激活函数为f_h（缺省是logistic函数），偏倚为b_h($|H|<|A|$)。
3. 输出层：$|A|$个softmax型节点或归一化指数节点，带有下标X及偏倚b_X。
4. 连接：$\alpha=$（α_{hi}：输入点i到隐节点h的连接），$\beta=$（β_{Xh}：隐节点h到输出节点X的连接），这与HMM前向或后向变量不会发生混淆。

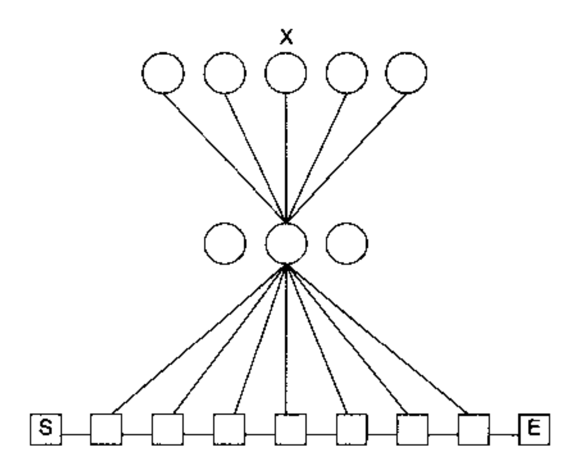


图9-10　简单混合结构（HMM状态的图示）

每一状态到公共隐层是全连接的。隐层中的每一节点全连接到每一个归一化指数输出节点。每个输出节点计算生成概率值e_{iX}。

对于输入i，隐层中第h个节点的活性由下式给定

$$f_h(\alpha_{hi}+b_h) \tag{9.3}$$

在输出层中对应的输出是

$$e_{iX}=\frac{e^{-\left[\sum_h \beta_{Xh} f_h(\alpha_{hi}+b_h)+b_X\right]}}{\sum_Y e^{-\left[\sum_h \beta_{Yh} f_h(\alpha_{hi}+b_h)+b_Y\right]}} \tag{9.4}$$

注　释

关于混合HMM/NN结构，许多地方值得注意：

- HMM状态可以分割为几个组，不同的组带有不同的网络。在蛋白质研究中，对于插入状态和主状态，或者对于蛋白质序列中对应于不同区域（疏水区、亲水区、α螺旋等）的不同组，可以用不同的神经网络。
- 利用带有$|H|$个隐节点的小隐层，很容易减少HMM参数，其中，$|H|$比N和$|A|$小。在图9-10中，带有$|H|$个隐节点且仅考虑主状态，混合HMM/NN结构的参数个数为$|H|$（$N+|A|$），而对应的简单HMM中参数个数为$N|A|$。对于蛋白质模型，混合HMM/NN结构能粗略地导出需要$|H|N$个参数，而对应的简单HMM有$20N$个参数。$|H|=|A|$与（7.33）大致相当。
- 参数数目可以通过改变隐节点的数目来进行合适的调整，以适应各种训练集的大小。在数据库的大小经常变动的环境下，这一点是有用的，目前它在分子生物学中的应用正是如此。
- 整个神经网络技术，如径向基函数、多隐层、稀疏连通性、权值共享、高斯先验分布以及超参数等，都可以用于网络构建。各种初始化和网络结构可以以灵活的方式实现。通过将不同的隐节点数分配到不同的生成或转移子集，必要时可以很容易地在模型中支持特定的通道类别。在图7-2的HMM中，在任何学习之前，我们通常必须使模型更倾向于主状态而非插入状态。利用权值共享和其他类型的长程相关性，也容易将可能相似的蛋白质区域连接在一起。通过合理设置输出偏倚的值，模型可以初始化为训练序列的平均组成或者任意其他有用的分布。
- 容易加入替代矩阵形式的先验信息。替代矩阵（见参考文献［8］、第1章和第10章）可以根据数据库计算得到，它本质上得到了一个背景概率矩阵$P=(p_{XY})$，其中p_{XY}是经过一定的进化时间，X将转变为Y的概率。矩阵P可以通过生成神经网络中的一个线性变换加以实现。
- 虽然带有连续生成分布的HMM的内容已经超出本书的讨论范围，但这类HMM的讨论也可以并入HMM/NN讨论框架。输出生成分布可以用样本、矩、混合系数等形式表示。在经典的高斯型混合情形下，期望、协方差和

混合系数可以通过神经网络计算得到。类似地，附加的HMM参数，例如用于模拟在任意给定状态下的驻留持续时间的指数参数可以通过一个神经网络计算得到。

简单HMM/NN结构的表示

考虑上面（图9-10）所描述的特殊HMM/NN结构，其HMM状态的一个子集完全连接到$|H|$个隐节点，并且这些隐节点完全与$|A|$个softmax输出节点连接。由于对任意的HMM状态i、任意的偏倚向量b_h以及任意的连接向量α_{hi}，存在一个新的连接向量α'_{hi}，这个新向量可生成相同的0偏倚的隐节点输出向量。在上述意义上，隐节点偏倚实际上不是必要的。这点在一般情况下不成立，例如一旦存在多隐层或输入节点到隐层不是完全内部连接，这点就不正确了。从一般性出发，我们保留了偏倚。而且即使不能扩大可表示的空间，偏倚仍可以使学习过程更方便。对于激活函数，类似的性质在一般情况下也成立。对于一个输入层与一个单隐层完全连接的情况，相同的隐层输出可以通过调整权重值由不同的激活函数来实现。

一个自然产生的问题是：如何表示隐层？在这种方式下可以实现的生成分布空间是什么？网络中每一个HMM的状态可以通过超立方体（hypercube）$[-1,1]^{|H|}$中的一个点来表示。点的坐标是$|H|$个隐节点的输出值。通过改变到隐节点的连接，一个HMM状态可以在超立方体中占据任意的位置。因此，可以实现的生成分布空间完全由从隐层到输出层的连接确定。如果这些连接保持固定不变，则每个HMM状态可以在超立方体中选择一个相应的最优点，在那点上，由神经网络权重值生成的生成分布最接近于真实的最优分布。在在线学习的过程中，所有的参数被同时学习，因此可以考虑其他影响因素。

为了进一步理解可实现的分布空间，需要考虑从隐层到输出节点的转化。为概念阐述上方便起见，我们引入一个附加的隐节点，它的赋值总为1，标号为0，用以表示形式为$b_X=\beta_{X0}$的输出偏倚。如果在这个扩展的隐层中轮流将每个隐节点的值设置为1，我们在输出层$p^h=(p_X^h)(0\leqslant h\leqslant|H|)$中得到$|H|+1$个不同的生成分布，其中

$$p_X^h=\frac{e^{-\beta_{Xh}}}{\sum_{Y\in A}e^{-\beta_{Yh}}} \tag{9.5}$$

现在考虑形式为（$1,\mu_1,\cdots,\mu_{|H|}$）的隐层中一般的输出模式。利用（9.4）和（9.5），输出层的生成分布是

$$e_{iX}=\frac{e^{-\sum_{h=0}^{|H|}\beta_{Xh}\mu_h}}{\sum_{Y\in A}e^{-\sum_{h=0}^{|H|}\beta_{Yh}\mu_h}}=\frac{\prod_{h\in H}\left[p_X^h\right]^{\mu_h}\left[\sum_{Y\in A}e^{-\beta_{Yh}}\right]^{\mu_h}}{\sum_{Y\in A}\prod_{h\in H}\left[p_Y^h\right]^{\mu_h}\left[\sum_{Z\in A}e^{-\beta_{Zh}}\right]^{\mu_h}} \quad (9.6)$$

化简后得到

$$e_{iX}=\frac{\prod_{h\in H}\left[p_X^h\right]^{\mu_h}}{\sum_{Y\in A}\prod_{h\in H}\left[p_Y^h\right]^{\mu_h}} \quad (9.7)$$

因此，所有通过神经网络实现的生成分布有（9.7）的形式，而且生成分布可以看做与每一隐节点相联系的$|H|+1$个基本分布P^h的组合。一般地，这种组合与P^h的凸线性组合不同。这种组合由三步操作组成：（1）对P^h的每个元件取μ_h次方，得到的h个隐节点的输出；（2）相应向量的所有组成元件相乘；（3）归一化。在这种形式下，混合HMM/NN处理方法与Dirichlet分布混合的方法不同。

学　习

HMM/NN结构可以用ML或MAP估计进行优化。与HMM不同，对于混合HMM/NN结构，EM算法的M步一般不能解析地实现。然而，我们仍可以利用一些基于链式法则的梯度下降法计算似然函数对HMM参数的偏导数，以及计算HMM参数对神经网络参数的偏导数。计算过程中，容易加入先验概率项的求导结果。通过只利用最有可能的路径，还可以使用Viterbi学习算法。学习方程的推导留做练习，这些也可以在参考文献［40］中找到。在最后得到的学习方程中，HMM动态规划和神经网络反向传播的成分紧密地联系在一起。这些算法也可以看做GEM（广义最大似然估计）算法。

9.5.1　多模型情形

上面描述的混合HMM/NN结构处理了HMM的第一个局限性：模型结构和复杂度的控制。不管神经网络元件如何复杂，最终模型仅仅是一个单HMM。因此HMM的第二个局限性，即长程相关性仍然没有得到解决。这个难题不能简单地通过采用高阶HMM来克服。最常见的障碍是高阶HMM在计算上难以处理。一种可能的处理方法是通过对每个相关上下文关系引入新状态，设法建立带有可变记忆长度的马尔可夫模型。这要求设计一种系统化的方法，这种方法直接从数据中确定可变长度的上下文关系。此外，我们必须希望这种处理保持小的相关上下文的数目。依照此思路，参考文献［448］给出了一个有趣的处理方法。这种方法

用直到10个字符左右的可变记忆长度马尔可夫过程，建立了一个处理英文的模型。

为了不求助于不同的模型类来处理第二个局限性，我们必须考虑更加一般的HMM/NN混合结构，其中的基本统计模型是一系列HMM。为了理解这一结构，再一次考虑在第8章的最后提到的X–Y/X′–Y′问题。捕获这种相关性要求在相应位置上存在可变生成向量和连接机制。在简单的情况下，必须用四个不同的生成向量，即e_i，e_j，e_i'和e_j'。这些向量中的每一个必须为字符X，Y，X′和Y′分配高的概率值。更重要的是，必须有一些具有记忆能力的向量，它们有一种连接点i和j的分布的机制，使得e_i和e_j用于序列O，而e_i'和e_j'用于序列O'。e_i和e_j'（或e_i'和e_j）的结合应该很少或不允许，除非数据要求如此。因此，e_i和e_j必定属于第一个HMM，而e_i'和e_j'必定属于第二个HMM，而HMM之间相互转移的概率是输入序列的函数。另外一种方法是必须使用一个带有可变生成分布的单HMM，用一些输入来调节它。

然而，在以上两种情况下，我们都认为给定状态的生成分布不仅依赖于它自己的状态，还依赖于一种附加的信息流I，此时有$\theta=f(i, I)$。同样地，在一个多重HMM/NN混合结构中，这个更复杂的函数f可以由神经网络计算得到。依赖于具体问题，输入I可以假设成不同的形式，可以称为“上下文”或“潜在”的变量。在可行的时候，I甚至可以等同于实时观察序列O。然而，其他输入取不同的字符集是可能的。在蛋白质建模中，一个明显的候选方案将是蛋白质二级结构（α螺旋、β折叠和无规卷曲）。一般地，I也可以是任何其他数组，代表调节HMM的潜在变量（latent variable）。[374] 我们简要地考察两个例子。

HMM专家模型的混合

第一种可能的处理方法是考虑模型M，此模型是n个简单的隐马氏模型$M_1,\cdots,M_n$的混合分布。（2.23）对任意的序列O，有

$$\mathbf{P}(O|M)=\sum_{i=1}^{n}\lambda_i\mathbf{P}(O|M_i) \tag{9.8}$$

其中混合系数λ_i满足$\lambda_i\geq 0$和$\sum_i\lambda_i=1$。在生成模式中，序列通过每一个独立的HMM随机产生，选中M_i的概率为λ_i。这样一种系统可以看做一个较大的单HMM，它的起始状态与HMM中的每一个M_i以转移概率λ_i进行连接（图8–5）。正如我们在第8章中所见到的，为了对球蛋白序列进行非监督分类，参考文献［334］中使用了这类模型。注意每一个子模型的各个参数可以通过神经网络计算得到一个

HMM/NN混合结构。由于HMM专家模型构成一个更大的单HMM，因此相应的混合结构等同于9.2节中的模型。HMM专家模型的独有特性是：此时的状态已经被复制和分群，以便构建不同的子模型。下一个步骤是得到可变的混合系数，这些混合系数依赖于输入序列或其他一些相关信息。这些混合系数可以计算出来作为神经网络的softmax节点输出，就像参考文献［277］中的混合专家模型构建那样。

生成专家模型的混合

考虑到生成参数e_{iX}应该也是附加输入I的函数，另一种处理方法是调节一个单HMM。因此有$e_{iX}=P(i, X, I)$。不失一般性，我们假设P是n个生成专家模型P_j的混合：

$$P(i, X, I)=\sum_{j=1}^{n}\lambda_j(i, X, I)P_j(i, X, I) \qquad (9.9)$$

在许多有意思的情形下，λ_j独立于X，结果有字符集上的概率向量方程

$$P(i, I)=\sum_{j=1}^{n}\lambda_j(i, I)P_j(i, I) \qquad (9.10)$$

如果$n=1$且$P(i, I)=P(i)$，则回到了一个单HMM。通过进一步假设λ_j不依赖于i，以及$P_j(i, X, I)$不直接依赖于I，就导出了一个重要的特殊情形，即

$$P(i, I)=\sum_{j=1}^{n}\lambda_j(I)P_j(i) \qquad (9.11)$$

这就为设计一般HMM/NN混合结构的顶层提供了一种原则性方法，例如在图9-11中描绘的结构。

分布P_j由神经网络计算得到，混合系数由另一条神经网络途径计算得到。自然，有可能发生许多变化；而且在最一般的情形下，切换网络可以取决于状态i，分布P_j取决于输入I。在蛋白质建模中，如果切换网络依赖于状态i，生成专家模型就对应于不同的区域类型，如疏水性区域和亲水性区域，而不是对应于蛋白质家族中的不同子类。

学 习

对于所有给定参数、一个给定的观察序列和输入向量I，一般的HMM/NN混合结构简化为一个单HMM。一个序列关于此HMM的似然度或者其他一些序列拟合水平，都可以通过动态规划计算获得。只要似然度对模型参数是可微的，就可

以通过神经网络，包括依赖于I的那部分网络（例如图9-11中的控制网络部分）反向传播梯度。做些微小的调整，我们就将导出学习算法，这种学习算法类似于上面描述过的那些算法。这种类型的学习算法支持图9-11所示的生成专家模型系统内部的协同。在通常的专家混合体系结构中，在各专家系统之间引入一定水平的竞争度也许有用，由此可使每一专家系统专门分析不同的序列子类。[277]

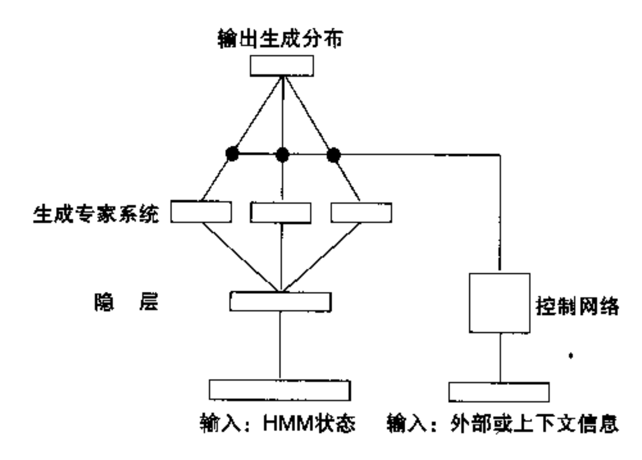


图9-11 一般HMM/NN的结构图示

其中的HMM参数由一个任意复杂度的神经网络计算，此神经网络操作状态信息，也操作输入或上下文信息。输入或上下文信息通过诸如切换和混合不同参数的专家系统来调整HMM参数。为简单起见，图中仅表示了有三个生成专家系统和一个单隐层的生成参数。从HMM状态到控制网络的连接，以及从输入到隐层的连接也是可能的。

当输入空间已经选定而相关输入I的值未知的时候，利用贝叶斯反演，可以通过学习得到输入的值和模型参数。考虑有一个输入I和每个观察序列O相关，而且混合模型参数为w的情形，我们可以计算$\mathbf{P}(O|I, w)$。将$\mathbf{P}(I)$和$\mathbf{P}(w)$分别定义为关于I和w的先验概率，则有

$$\mathbf{P}(I|O, w)=\frac{\mathbf{P}(O|I, w)\mathbf{P}(I)}{\mathbf{P}(O|w)} \tag{9.12}$$

其中

$$\mathbf{P}(O|w)=\int_I \mathbf{P}(O|I, w)\mathbf{P}(I)\, dI \tag{9.13}$$

给定数据后，利用贝叶斯定理，模型参数的概率可以由下式计算得到

$$\mathbf{P}(w|D)=\frac{\mathbf{P}(D|w)\,\mathbf{P}(w)}{\mathbf{P}(D)}=\frac{\left[\prod_O \mathbf{P}(O|w)\right]\mathbf{P}(w)}{\mathbf{P}(D)} \tag{9.14}$$

此时我们假设各观测值之间是独立的。这些参数可以通过$-\log\mathbf{P}(w|D)$的梯度下降法进行优化。其主要的步骤是估计似然度$\mathbf{P}(O|w)$及其关于w的导数，这些可以通过蒙特卡罗采样实现。潜在变量I的分布可以由（9.12）计算。参考文献［374］给出这种学习方法的一个例子。用于蛋白质建模的密度网络（density network）在本质上可以视为HMM/NN混合结构的一种特殊情形，其中的每一生成向量都可以视为对低维空间上真实的“隐”输入I（I和w具有独立的高斯先验分布）进行的一种softmax变换。输入I按照序列的函数调节生成向量，进而调节那个基本的HMM。

9.5.2　仿真结果

现在我们利用免疫球蛋白家族研究的例子，[40]考察HMM/NN单模型混合结构原则的简单应用。免疫球蛋白或抗体是由B细胞产生的蛋白质。抗体能特异性地结合外来抗原，从而使抗原被中和或被其他效应细胞破坏。各类不同的免疫球蛋白由它的轻链和重链对确定，轻链和重链主要通过二硫键结合在一起。每条轻链和重链分子都包含一个可变区（V）和一个（对于轻链）或多个（对于重链）不变区（C）（图9-12）。V区在各种免疫球蛋白之间是不同的，它可以识别特异抗原。V区有大约1/3的氨基酸形成超可变位点，这些位点负责脊椎动物免疫应答的多样性。这里所用的数据库与参考文献［41］中所用的数据库相同，该数据库由人和小鼠的重链免疫球蛋白的V区序列组成，它们都取自PIR数据库。所用数据有224条序列，最小长度是90，最大长度是254，平均长度N为117。

免疫球蛋白的V区首先用单HMM建模，[41]它总共含有$52N+23=6\ 107$个可调节的参数，这一模型类似于图7-3中的一个。这里我们考虑一个具有以下特征的混合HMM/NN结构。其基本模型是一个具有图7-3的结构的HMM。所有主状态生成通过带有2个隐节点的公共神经网络计算得到。同样地，所有插入状态生成通过带有1个隐节点的公共神经网络计算得到。每个状态转移分布通过一个不同的softmax网络计算获得。忽略边界效应，这个HMM/NN结构的参数总个数为1 507：其中有（$117\times3\times3$）=1 053个转移参数，（$117\times3+3+3\times20+40$）=454个生成参数，其中包括偏倚。这个体系结构只以演示为目的，没有进行优化。我们估计它的参数数目还可以进一步减少。

然后利用梯度下降法和相应的Viterbi学习算法对这一混合结构进行在线训练。

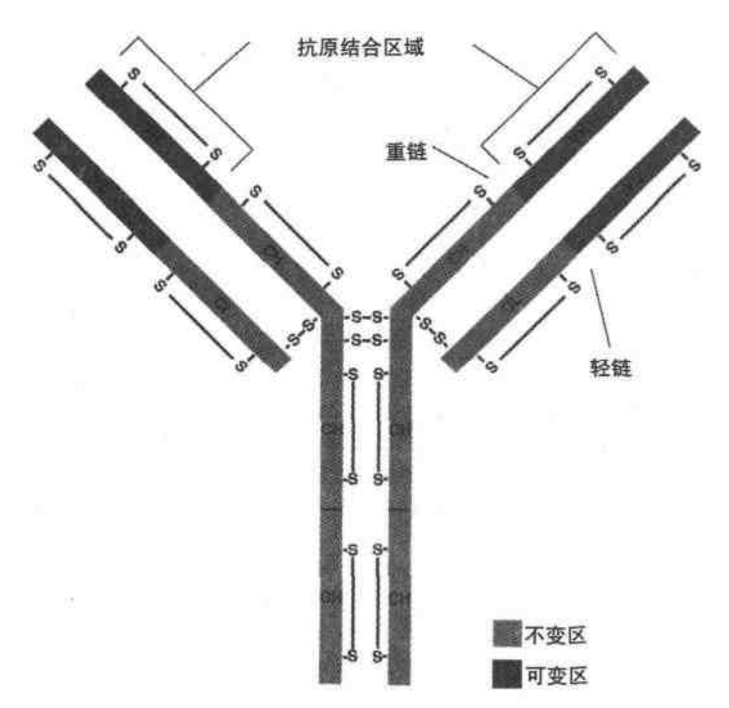


图9-12 由两条轻多肽链（L）和两条重多肽链（H）组成的典型人类抗体分子的结构模型图

图中标明了链间和链内的二硫键（S）。半胱氨酸残基由二硫键连接。对应于不同区域，抗原结合的两个相同的活性位点定位在抗体分子的双臂上。

训练集由150个序列的随机子集组成，与用于简单HMM实验的训练集一致。所有从输入到隐层的权重值都用独立的高斯分布初始化，这一高斯分布数学期望为0，标准差为1。所有从隐层到输出层的权重值都初始化为1。这样在所有生成状态中就导出了一个均匀的生成概率分布。[⑨]注意：如果所有权重值都初始化为1，包括那些从输入层到隐层的权重值，则隐节点不能与其他节点区分。移出插入或删除状态的转换概率一率初始化为1/3。然而，我们沿主线引入一个小偏倚，以非对称Dirichlet先验分布的形式支持从主状态到主状态的转移。这一先验概率等价于在目标函数中引入一个正则项，这里的目标函数等于主线转换路径值的对数。将调整常数设置为0.01，学习率设置为0.1。一般地，要达到平衡，经过10步训练循环已经足够了。

在图9-13中，我们演示了20条随机地选自训练集和校验集的免疫球蛋白序列的多重序列比对的结果。校验集由剩余的74条序列组成。这个比对结果在5到10个训练周期之间非常稳定。它对应于用Viterbi学习算法训练了10个周期的模型。这

⑨ 就Viterbi学习算法来说，这样的处理，例如平均组分初始化处理，可能优于非均匀初始化处理，因为非均匀初始化会在Viterbi路径中引起变形。

```
                               1                      24
F37262  ---------------------------AELM--KPGASVKISCKATG--YKFSS----Y--------WIEWVKQRPGHGLEWIGENL-
B27563  ----------------------LQQPGAELV--KPGASVKLSCKASG--YTFTN----Y--------WIHWVKQRPGRGLEWIGRID-
C30560  -------------------QVHLQQSGAELV--KPGASVKISCKASG--YTFTS----Y--------WMNWVKQRPGQGLEWIGEID-
G1HUDW  -------------------QVTLRESGPALV--RPTQTLTLTCTFSG--FSLSGetmc----------VAWIRQPPGEALEWLAWDI-
S09711  mkhlwfffllvraprwclsQVQLQESGPGLV--KPSETLSVTCTVSG------------gsvsssglYWSWIRQPPGKGPEWIGYIY-
B36006  ---------------------------------------KISCKGSG--YSFTS----Y--------WIGWVRQMPGKGLEWMGIIY-
F36005  -------------------QVQLVESGGGVV--QPGRSLRLSCAASG--FTFSS----Y--------AMHWVRQAPGKGLEWVAVIS-
A36194  mgwsfiflfllsvtagvhsEVQLQQSGAELV--RAGSSVKMSCKASG--YTFTN----Y--------GINWVKQRPGQGLEWIGYQS-
A31485  -------------------EVKLDETGGGLV--QPGRPMKLSCVASG--FTFSD----Y--------WMNWVRQSPEKGLEWVAQIRN
D33548  -------------------QVQLVQSGAEVK--KPGASVKVSCEASG--YTFTG----H--------YMHWVRQAPGQGLEWMGWIN-
AVMSJ5  -------------------EVKLLESGGGLV--QPGGSLKLSCAASG--FDFSK----Y--------WMSWVRQAPGKGLEWIGEIH-
D30560  -------------------QVQLKQSGPSLV--QPSQSLSITCTVSD--FSLTN----F--------GVHWVRQSPGKGLEWLGVIW-
S11239  melglswifllailkgvqcEVQLVESGGGLV--QPGRSLRLSCAASG--FTFND----Y--------AMHWVRQAPGKGLEWVSGIS-
G1MSAA  -------------------EVQLQQSGAELV--KAGSSVKMSCKATG--YTFSS----Y--------ELYWVRQAPGQGLEDLGYIS-
I27888  -------------------EVQLVESGGGLV--KPGGSLRLSCAASG--FTFSS----Y--------AMSWVRQSPEKRLEWVADIS-
PL0118  ---------------------QLQESGSGLV--KPSQTLSLTCAVSGgsISSGG----Y--------SWSWIRQPPGKGLEWIGYIY-
PL0122  -------------------EVQLVESGGGLV--QPGGSLKLSCAASG--FTFSG----S--------AMHWVRQASGKGLEWVGRIRS
A33989  -------------------DVQLDQSESVVI--KPGGSLKLSCTASG--FTFSS----Y--------WMSWVRQAPGKGLQWVSRISS
A30502  -------------------EVQLQQSGPELV--KPGASVKMSCKASG--DTFTS----S--------VMHWVKQKPGQGLEWIGYIN-
PH0097  -------------------DVKLVESGGGLV--KPGGSLKLSCAASG--FTFSS----Y--------IMSWVRQTPEKRLEWVATIS-

             60        70        80        90      100
F37262  -P-G-SDSTKYNEKFKGKATFTADTSSNTAYMQLSSLTSEDSAVYYCARnyygssnlfay-----------------------------
B27563  -P-N-SGGTKYNEKFKNKATLTINKPSNTAYMQLSSLTSDDSAVYYCARgydysyya-------------MDYWGQGTsvtvss---
C30560  -P-S-NSYTNNNQKFKNKATLTVDKSSNTAYMQLSSLTSEDSAVYYCARwgtgsswg------------WFAYWGQGTlvtvsa---
G1HUDW  -----lNDDKYYGASLETRLAVSKDTSKNQVVLSMNTVGPGDTATYYCARscgsq---------------YFDYWGQGIlvtvss---
S09711  ---Y-SGSTNYNPSLRSRVTISVDTSKNQFSLKLGSVTAADTAVYYCARvlvsrtsisqysy-------YMDVWGKGTtvtvss---
B36006  -P-G-DSDTRYSPSFQGQVTISADKSISTAYLQWSSLKASDTAMYYCARrrymgygdqa-----------FDIWGQGTmvtvss---
F36005  -Y-D-GSNKYYADSVKGRFTISRDNSKNTLYLQMNSLRAEDTAVYYCAR--------------DRKASDAFDIWGQGTmvtvss---
A36194  -T-G-SFYSTYNEKVKGKTTLTVDKSSSTAYMQLRGLTSEDSAVYFCARsnyyggsys-------------FDYWGQGTtltvss---
A31485  KP-Y-NYETYYSDSVKGRFTISRDDSKSSVYLQMNNLRVEDMGIYYCTGsyyg------------------MDYWGQGTsvtvss---
D33548  -P-N-SGGTNYAEKFQGRVTITRDTSINTAYMELSRLRSDDTAVYYCARasycgydcyy-----------FFDYWGQGTlvtvss---
AVMSJ5  -P-D-SGTINYTPSLKDKFIISRDNAKNSLYLQMSKVRSEDTALYYCARlhyygyn-----------------AYWGQGTlvtvsae--
D30560  -P-R-GGNTDYNAAFMSRLSITKDNSKSQVFFKMNSLQADDTAIYYCTKegyfgnydya-----------MDYWGQGTsvtvss---
S11239  --wD-SSSIGYADSVKGRFTISRDNAKNSLYLQMNSLRAEDMALYYCVKgrdyydsggyftva--------FDIWGQGTmvtvss---
G1MSAA  -S-S-SAYPNYAQKFQGRVTITADESTNTAYMELSSLRSEDTAVYFCAVrvisryfdg-----------------WGQGTlv-------
I27888  -S-G-GSFTYYPDTVTGRFTISRDDAQNTLYLEMNSLRSEDTAIYYCTRdeedpttlvapfa---------MDYWGQGTsvtvs----
PL0118  ---H-SGSTYYNPSLKSRVTISVDRSKNQFSLKLSSVTAADTAVYYCAR----------------------------------------
PL0122  KA-N-SYATAYAASVKGRFTISRDDSKNTAYLQMNSLKTEDTAVYYCTR----------------------------------------
A33989  KA-D-GGSTYYADSVKGRFTISRDNNNNKLYLQMNNLQTEDTAVYYCTRearwggw--------------YFEHWGQGTmvtvts---
A30502  -P-Y-NDGTKYNEKFKGKATLTSDKSSSTAYMELSSLTSEDSAVYYCARgg----------------------FAYWGQGTlvtv-----
PH0097  -S-G-GRYTYYSDSVKGRFTISRDNAKNTLYLQMSSLRSEDTAMYYSTAsgds--------------------FDYWGQGTtltvssak-
```

图9-13 从训练集和校验集中随机选取20个免疫球蛋白的多重序列比对结果

序列：F37262、G1HUDW、A36194、A31485、D33548，S11239、I27888、A33989和A30502。比对由一个混合HMM/NN结构经过10个周期的训练得到，此混合HMM/NN结构对于主状态生成带有2个隐节点，对于插入状态生成带有1个隐节点。其中小写字母代表插入状态的输出。注意在模型中，一些序列中的信号肽由于在第一个插入状态中反复过渡而被捕获。

一比对与先前用简单HMM导出的多重序列比对方法类似，只是现在方法的参数数目为先前方法的4倍多。这种算法已经可以发现大部分显著的家族特征。最重要的是，接近于区域开始和结尾部分的半胱氨酸残基（C）（多重序列比对中的位点10和100）对准得非常完美，这些残基是形成结合两链的二硫键的原因。仅有的例外是最后一个序列（PH0097），在这个序列的末端部分有一个丝氨酸残基（S）。这种情况是罕见的，被认为是该位点保守性的例外。数据集中的一部分序列在N端出现一个信号肽序列（见6.4节）。在训练之前，我们不去掉它们。通过把信号肽处理为初始重复插入，模型可以探测和适应它们，正如从三个序列（S09711、A36194、S11239）比对中看到的那样。这个多重序列比对算法也有一些孤立的问题，这些问题和过分使用间隙和插入状态有部分关系。有趣的是，这些情况在超可变区域最明显，例如在位置30至35，以及50至55之间的区域。这些问题应该通过更精心地选择混合结构和（或）正则化加以消除。在这种情况下的比对，用梯

度下降法和（或）多达4个以上的隐节点似乎也没能改善性能。

在图9-14中，我们给出了与每个主状态相关联的2个隐节点的输出图。对于大多数状态，至少1个输出是饱和的。组成二硫键的半胱氨酸残基（主状态24和100），其2个节点的输出都是饱和的，而且在相同的区域（-1,+1）中。接近中心（0,0）的点对应仅仅由偏倚确定的生成分布。

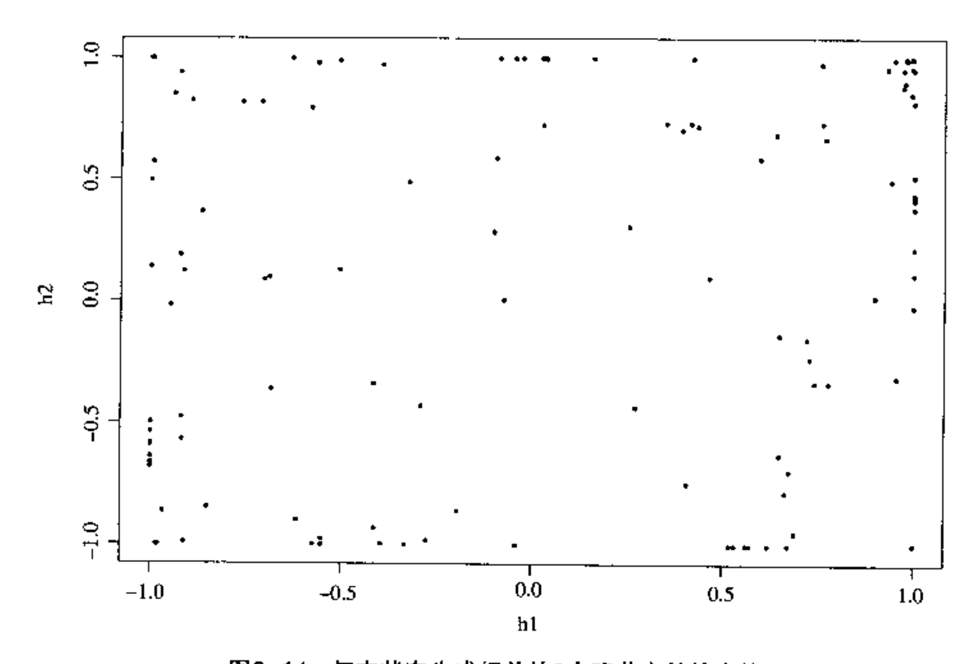


图9-14 与主状态生成相关的2个隐节点的输出值

与半胱氨酸（C）相关的2个输出值位于左上角，几乎与坐标（-1,+1）重合。

9.5.3 总 结

我们已经描述了一大类混合HMM/NN结构。这些结构在两个方面改进了单HMM的性能：第一个方面，神经网络的重新参数化为控制模型复杂度、引入先验值、建立利用输入调节最终模型的机制提供了一个灵活的工具；第二个方面，利用多个HMM建模可以覆盖一类更大的分布集，表达非平稳性和相关性等单HMM无法处理的问题。类似的思想在参考文献［58］中用输入—输出HMM（IOHMM）的概念加以引入。HMM/NN处理方法是对已有的在序列模型中合并先验信息的技术的补充，而不是其替代物。

为了成功地将混合HMM/NN结构应用在实际问题中，需要处理的两个重要问

题是神经网络结构的设计和外部输入或上下文信息的选择。这些依赖于所处理的问题，没有通用的处理方法。我们已经描述了一些构架的例子，如利用混合的思想设计神经网络元件。也可以选择不同的输入，例如选择上下文信息、不同字符集的序列或连续参数变量。[374]

这一部分所描述的方法不只限于HMM，这些方法可以应用到任何概率模型。其基本思想是利用神经网络（或其他任何灵活的重新参数化方法）计算或调节模型参数。几个混合结构的例子可以在参考文献［395］中找到（如参考文献［395］）。事实上，第5章中的神经网络结构可以看做混合构架。在标准回归的情况下，高斯模型用于输入空间的每一个点。每个高斯模型由它的均值参数化。标准的神经网络构架在每一点简单地计算这个均值。虽然混合建模的原理并不新颖，但是通过把这一原理系统地应用在HMM中，我们建立了新的一类模型。在其他模型类中，这一原理还没有被系统地应用，例如在进化概率模型（第10章）和随机文法模型（第11章）中。在下一节中，我们将遵循参考文献［37］的思路并将类似的技术应用到一大类概率模型中，即将其应用到BIOHMM和预测蛋白质二级结构的问题中。

9.6 用于蛋白质二级结构预测的双向反馈神经网络

蛋白质二级结构预测（也见6.3节）可以表示为学习序列同步转换的问题，这是从氨基酸字符集中的字符串到结构类别字符集中的字符串之间的转换。因为生物序列是一种空间序列，而不是时间序列，我们已经看到BIOHMM是一类很有意思的处理这一问题的新的图模型。特别地，BIOHMM给出了一种较好的方法，替代基于固定宽度输入窗的方法。这些模型的表达能力可以使它们捕获到远程信息，这些信息以上下文知识的形式储藏在隐状态变量中。依靠这种方式，这些模型可以潜在地克服前馈网络的主要不足之处，即随着窗口的增大，有关参数的数目呈线性增加。由于其稳定性（即参数不随时间变化）而造成的隐含的权重共享，因此直觉上看，这些模型的参数数目应该很少。

我们已经将BIOHMM直接应用到蛋白质二级结构预测中，并且取得了一些成功。[36] 然而作为图模型，BIOHMM含有无向环，因此要求使用计算量很大的信念传播（evidence-propagation）算法（交叉树算法[287]），而不是无环图中（如HMM）使用的简单的Pearl算法（见附录C）。因此为了加快算法的速度，我们可以使用前面章节中提到的技术，利用前馈和反馈神经网络技术，以此重新参数化模型。

9.6.1 双向反馈神经网络

以t定义蛋白质序列中的一个位置，总的模型可以视为一个概率模型。对于每一个t，模型输出一个向量$O_t=(o_{1,t}, o_{2,t}, o_{3,t})$，满足$o_{i,t}\geqslant 0$和$\sum_i o_{i,t}=1$。这些$o_{i,t}$是二级结构类的隶属概率。输出预测有如下形式：

$$O_t=\eta(F_t, B_t, I_t) \tag{9.15}$$

它依赖于前向（上游）信息F_t，后向（下游）信息B_t，以及在时刻t的输入I_t。向量$I_t\in \boldsymbol{R}^k$表示编码时刻t时的外部输入。最简单的情形是输入只限于一个单氨基酸，k=20，以及利用正交二元编码（见6.1节）。在这种情形下，没有必要加入一个额外的输入符号代表蛋白质的末端部分。扩展到几个氨基酸的大窗口当然也是可能的。函数η通过一个神经网络$\mathcal{N}_\eta$实现（见图9-15的中心和上端的连接）。因此为了确保一个一致的概率解释，网络$\mathcal{N}_\eta$的三个输出节点按照以下归一化指数形式（或softmax函数）获得

$$o_{i,t}=\frac{\exp\left(net_{i,t}\right)}{\sum_{l=1}^{3}\exp\left(net_{l,t}\right)} \quad i=1,2,3 \tag{9.16}$$

其中$net_{i,t}$是在位点t的第i个输出节点的输出值。模型的性能可以用估计分布和目标分布之间的相对熵来评估。

模型的新颖之处在于包含了向量$F_t\in \boldsymbol{R}^n$，尤其是$B_t\in \boldsymbol{R}^m$中的上下文信息。这些满足下列的双向反馈方程：

$$F_t=\phi(F_{t-1}, I_t)$$

$$B_t=\beta(B_{t+1}, I_t) \tag{9.17}$$

这里，$\phi(\cdot)$和$\beta(\cdot)$是可学习的非线性状态转移函数。它们可以在不同的形式下实现，但这里我们假设它们用两个神经网络——$\mathcal{N}_\phi$和$\mathcal{N}_\beta$（图9-15中的左子网络和右子网络）来实现，这两个神经网络各自带有n和m个logistic输出节点。因此，$\mathcal{N}_\phi$和$\mathcal{N}_\beta$分别为$n+k$和$m+k$个输入所馈送。特别是结合参考文献［445］中描述的权重共享方法后，实现大的输入窗口也是可能的，其中不同的输入可以用于计算F_t、B_t和O_t。前向链F_t储存了包含在时刻t之前的上下文信息，并与标准RNN中内部状态起着相同作用。模型的新部分以附加的后向链B_t的形式出现，负责储存包含在时刻t之后的上下文信息，即将来的信息。双向动态的实际形式由子

网络$\mathcal{N}_\phi$和$\mathcal{N}_\beta$间的连接权重控制。我们将看到，这些权重可以通过最大似然估计进行调整。由于（9.17）涉及两个再循环，必须给出序列开始和结束时候的两个相应的边界条件。为简单起见，这里我们取$F_0=B_{N+1}=0$，但是通过扩展参考文献［184］中建议的关于标准RNN的技术，使边界条件适应数据也是可能的。

离散时间指标t的范围从1到N，N是所考察的蛋白质的总长度。因此概率输出O_t通过一个RNN参数化，并且依赖于输入I_t和完整蛋白质序列中的上下文信息，此概率输出可以概括为向量对（F_t, B_t）。相反地，在一个通常的神经网络方法中，这些概率分布仅仅依赖于一个相对短的氨基酸子序列。直觉上我们可以将F_t和B_t想像为能沿着蛋白质被“滚动”的“轮子”。为了预测在位点t的类别，我们沿着从N端到C端的方向反向滚动“轮子”，直到位点t，然后结合在“轮子”中读到的内容和I_t，利用η计算出合适的输出。

从输入氨基酸序列到输出类别序列之间的全局映射，可以通过图9–16中的图模型来描述。这个网络代表变量I_t、F_t、B_t和O_t对于所有时间$t=1,\cdots,N$展开时的直接依赖性。每个节点由变量之一标号，弧线表示直接功能依赖性。除了I_t、F_t、B_t和O_t之间的内部关系在这里是确定的而不是概率性的外[(9.15)和(9.17)]，这个图表示基本的贝叶斯网络BIOHMM。然而，完整的BRNN模型是一个概率模型。正如我们所看到的，在BIOHMM中的推理是容易的，但是每一步的时间复杂度为$O(n^3)$（这里n为链中的典型状态数目），这一点限制了它们应用到二级结构的实际预测工作中。[36]

由（9.15）和（9.17）产生的结构显示在图9–15中，为简单起见，其中所有的NN都有一个单隐层。隐状态F_t被复制回输入。这些以图示形式出现在图9–15中，图中我们利用了因果移位操作子（causal shift operator）q^{-1}，这个操作子对类别的时间变量X_t进行操作，并且公式化地定义为$X_{t-1}=q^{-1}X_t$。类似地，q表示移位操作子q^{-1}的逆（或非因果复制），它由$X_{t+1}=qX_t$和$q^{-1}q=1$定义。如图9–15所示，一个非因果复制在隐状态B_t中被实现。明显地，移除｛B_t｝就回到了标准因果RNN。

模型自由度的数目依赖于两个因素：（1）前向和后向状态向量的维数n和m；（2）实现状态转移和输出函数的三个前馈网络中的隐节点的数目（见图9–15）。注意将BRNN规定为一个稳态网络这一点是重要的，即在网络实现中的连接权重值$\beta(\cdot)$、$\phi(\cdot)$和$\eta(\cdot)$不随时间变化，也就是不随蛋白质的位置变化。这是一种减少自由参数数目和过拟合风险的权重共享形式，它不必牺牲捕获远程信息的能力。

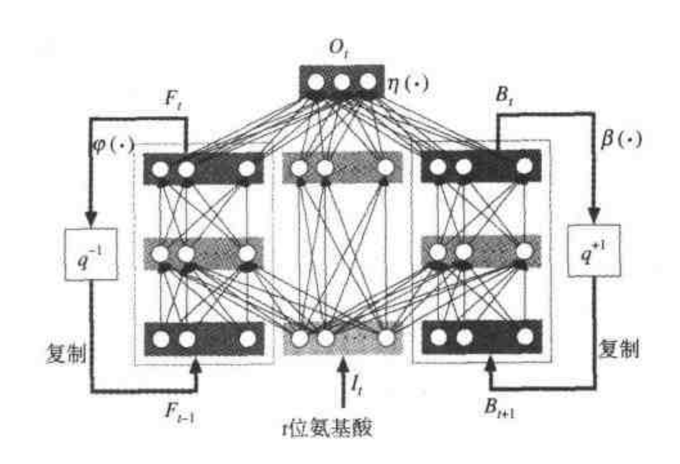


图9-15 双向反馈的神经网络结构

输入对应给定蛋白质序列中的氨基酸字符。输出对应相应的α螺旋、β折叠以及无规卷曲等二级结构类别。

9.6.2 推理和学习

由于图9-16中显示的图是无环图，因此可通过明确定义全局处理方案对其节点可进行拓扑排序。利用随时间展开的网络，BRNN预测算法从F_0=0开始，从左到右更新所有状态F_t。类似地，状态B_t从右到左进行更新。在前向和后向传播发生以后，预测O_t可以被计算出来。前向和后向传播仅需要对每个蛋白质序列进行一

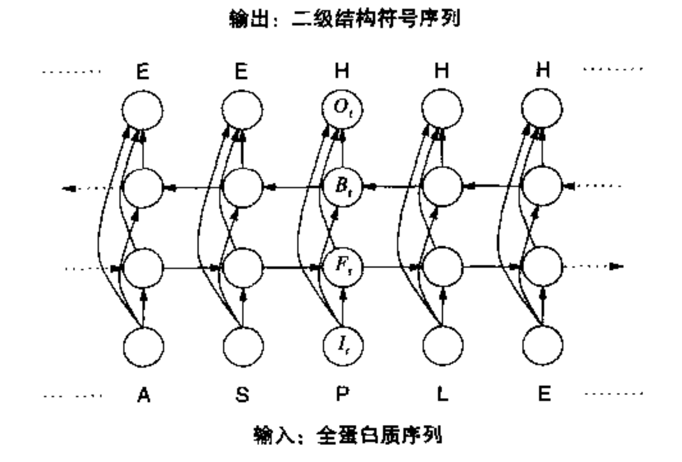


图9-16 双向BRNN中变量的直接相关性

边界条件由$F_0=B_{N+1}=0$以及与当前蛋白质序列相关的输入提供。

次从末端到末端的计算。因此，这种算法的时间复杂度是$O(NW)$，其中W是权重值的数目，N是蛋白质序列的长度。这个复杂度与由固定大小窗口馈给的前馈网络的复杂度相同。在BRNN的情形中，W以阶$O(n^2)$增加，通过在关于$\phi(\cdot)$和$\beta(\cdot)$的子网络中限制隐节点的数目，可以减少权重值的实际数目。因此，BRNN中的推理比BIOHMM中的推理更有效，BIOHMM中推理的复杂度为$O(Nn^3)$。[36]

学习问题可以表示为一个最大似然估计问题，其中，对数似然度本质上是在给定输入氨基酸序列后，二级结构的预测分布和真实条件分布之间的相对熵：

$$\ell = \sum_{\text{序列}} \sum_{t=1}^{N} z_{i,t} \log o_{i,t} \tag{9.18}$$

其中如果位置t的二级结构是i，则$z_{i,t}=1$，否则$z_{i,t}=0$。优化问题可以用梯度下降法解决。BRNN与标准的RNN之间仅有的区别在于它的梯度是通过考虑无因果瞬时相关性来计算确定的。由于散开型网络（unrolled network）是无环网络，广义的反向传播算法可以作为使用结构算法[188]的反向传播的一种特殊情形。从直觉上看，误差信号首先被赋到对应于输出变量O_t的叶节点（leaf node）上。然后，通过跟随散开型网络的任意反向拓扑类别，这一误差信号在两个时间方向上传播（见图9-16）。显然，这个步骤也涉及到经过NN隐层的反向传播。因为模型是稳态的，所以权重值在不同的时间点上、不同的NN复制之间是共享的。因此，通过对所有与不同时间步骤相关联的贡献值求和，就可以简单地获得整个梯度。

为了加快收敛速度，采用在线权重值更新方法很方便。一旦关于某个蛋白质的梯度已经被计算得到，权重值立即就可以更新。这个方法也可以通过增加一个启发式的学习率自适应算法来充实：如果在固定周期数目中的平均误差减少量降低到一个给定的阈值以下，则模型的学习率也随之降低。

9.6.3　长程相关性

在训练标准RNN时，主要的困难之一是梯度消失问题。[57]直觉上看，为了在时刻（或位置）t处对输出有所贡献，$t-\tau$时刻的输入信号必须在前向链中，通过神经网络的τ个复制来传播，以使执行状态转移功能。然而，在计算梯度时，误差信号必须沿着相同路径反向传播。每个传播都可以解释为误差向量与激活函数的Jacobian矩阵的乘积。遗憾的是，当动态模型形成允许系统可靠存储历史信息的吸引子（attractor）时，Jacobian矩阵的范数小于1。因此当τ比较大时，在时刻t，关于时刻$t-\tau$的输入的误差梯度以指数形式趋于消失。类似地，对于BRNN情形，在前

向链和后向链中的误差传播也以指数形式衰减。因此，虽然这种模型在原则上有能力存储远程信息，但这些信息不能被有效地学习。很明显，这是一个理论上的讨论，它的实际影响需要根据每种基本情形来评估。

在预测蛋白质二级结构的实践中，BRNN能够可靠地利用大约±15个氨基酸长度之内的输入信息（即总共有效的窗口大小大约是31个氨基酸）。它通过逐渐增加蛋白质片断长度来进行经验性估计并回馈给模型。我们观察到，如果蛋白质片断的长度变化超出41个氨基酸，中心残基位置的预测精度没有显著的变化。这是对输入窗口大小范围为11~17个氨基酸的标准神经网络的改进。[453,445,290] 据推测，在更长距离的地方也有相关的信息存在，但是到目前为止还不能找到它们。

为了限制这个问题，最近提出了一种补偿的方法。这种方法指出，梯度消失问题可以用一个针对输出的外部迟滞来缓和，它为误差信号的有效传播提供了短路径。[364] 遗憾的是，由于与双向传播结合的输出反馈在散开型网络中将形成环，因此这种思想方法不能直接应用到BRNN中。然而，一种类似的机制可以利用以下的调整动态过程进行补充：

$$F_t=\phi(F_{t-1}, F_{t-2}, \cdots, F_{t-s}, I_t)$$

$$B_t=\beta(B_{t+1}, B_{t+2}, \cdots, B_{t+s}, I_t) \qquad (9.19)$$

对于前向和后向状态的明显相关性，在图模型中引入了捷径连接，形成了梯度不能被传播的短路径。这与在概率模型情形中引入高阶马尔可夫链是相同的。然而，高阶马尔可夫链的参数个数以s的指数级别增长，而在这里，参数个数仅以s的线性增长，这点是不同的。为了减少参数个数，关于（9.19）的一种简化方法限制了对远离t的s残基状态向量的依赖性：

$$F_t= \phi(F_{t-1}, F_{t-s}, I_t)$$

$$B_t= \beta(B_{t+1}, B_{t+s}, I_t) \qquad (9.20)$$

这个基本结构的另外一种变化主要在于，在前向和后向状态链中馈送带有一个窗口的输出网络，以便增大有效的窗口长度。在这种情形中，预测值由下式计算得到：

$$O_t=\eta(F_{t-s},\cdots, F_{t+s}, B_{t-s},\cdots, B_{t+s}, I_t) \qquad (9.21)$$

注意，对于向量F_t和B_t，窗口可以在时刻t的过去和将来扩展。

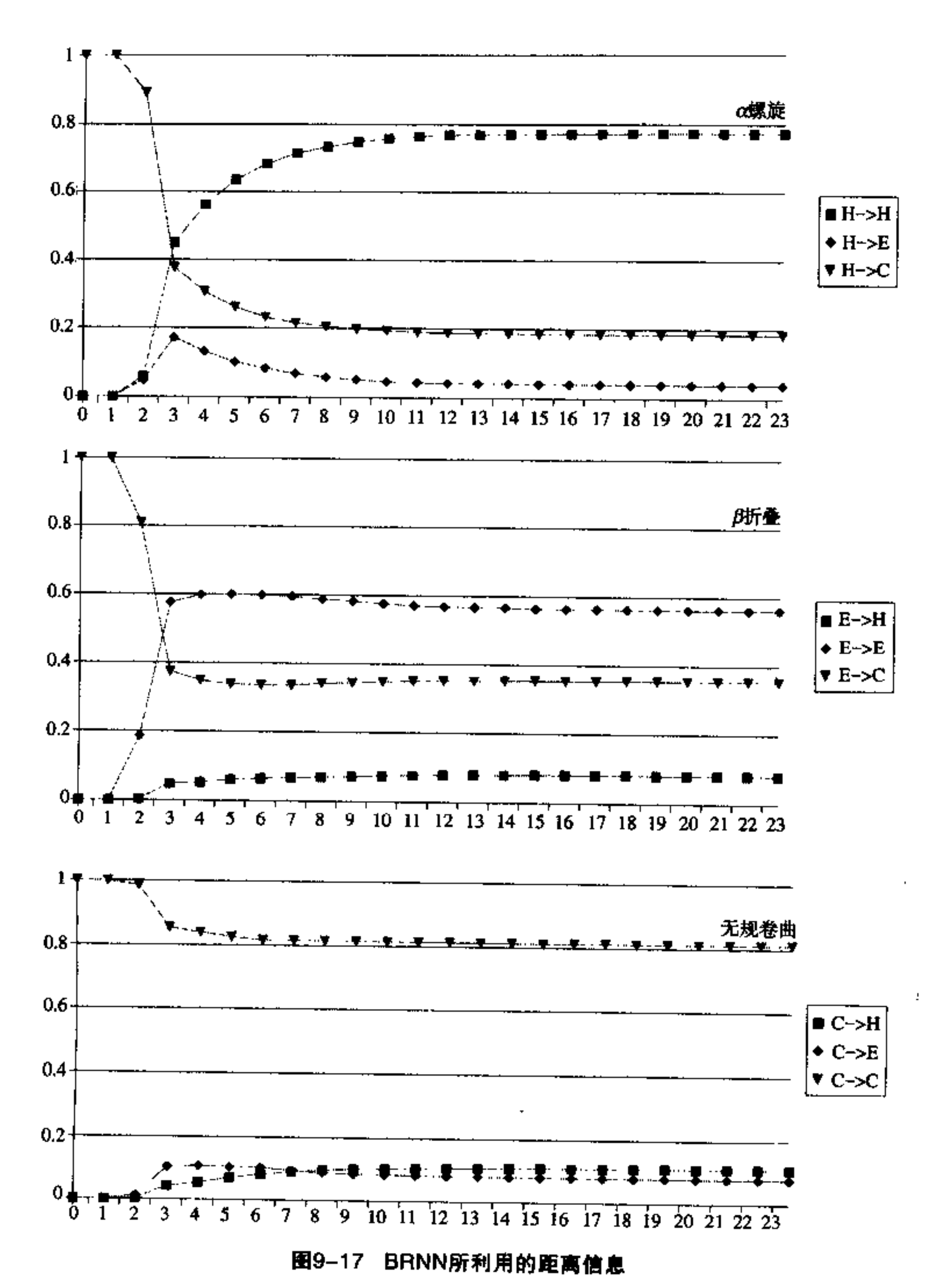


图9-17　BRNN所利用的距离信息

水平轴表示τ，即离给定位点的距离。在给定位点之后，所有的入口值均赋为0。每一条曲线表示测试集模糊矩阵的一个归一化行。

9.6.4 实现和结果

BRNN已经用于二级结构预测服务工具SSpro，此工具可以从因特网获得。[10]除了利用BRNN外，SSpro还利用了近年来已经证明对于预测二级结构有用的一些特征，例如总体特征和序列谱特征（见6.3节）。特别是在输入层上，序列谱特征被大量使用。第1版的SSpro使用了由BLAST生成的序列谱特征，它的实验细节和性能分析在见参考文献［37］。SSpro的最新版本使用了PSI-BLAST程序生成的序列谱特征，达到了大约80%的正确识别率。在2000年的CASP盲预测竞赛和Rost独立自动评估服务EVA（http://dodo.bioc.columbia.edu/~eva/）的测评结果中，SSpro已经进入最好的预测工具之列，而这些测评是基于每周存储在PDB数据库中的新序列进行的。

除了性能结果之外，为了研究BRNN模型捕捉长程信息的能力，研究人员进行了大量实验。对于每一个蛋白质和每一个氨基酸位点t，我们用0值代替所有在区间［$t-\tau$, $t+\tau$］之外的输入，并将获得的序列馈送给以上BRNN。对于0~23中不同的τ值，实验重复进行。结果显示在图9-17中，其中，每个图形中的曲线表示一个半窗口大小τ为0~23的测试集模糊矩阵的归一化行。例如在第一个图形中，作为τ的一个函数，由$H \rightarrow C$标号的线表示归类为蛋白质无规卷曲的百分率。当$\tau>15$时，曲线最稳定。虽然这个模型对很远距离的信息不敏感，但应该注意到，在相关的文献中，典型前馈网络都没有利用$\tau=8$之外的信息。

基因组和其他测序计划得到了大量蛋白质序列，因此二级结构预测精度提高哪怕是很小的百分率，对于结构基因组来说也是有意义的。到目前为止，结合图模型的机器学习算法和它们的神经网络参数化算法是这个研究领域最好的方法之一。这里提出的BRNN以及相关的思想开始处理长程相关性的问题。因此，现在BRNN已经发展到预测大量其他结构特征，包括β折叠中氨基酸配对数、相邻残基的数目和水溶性。[45,429]这些预测模块是蛋白质三维结构预测的更广泛的策略的一部分。这些策略基于接触图（contact map）的中间形态预测，始于初级序列及预测的结构特性，具有低（二级结构）和高（氨基酸）的分辨率。实际上，从三维空间中二级结构元件之间排列关系的预测到蛋白质拓扑结构和三维结构的预测，还需要走很长的路。

这项工作可以向几个方向拓展，其中包括构架上的许多变化。除了对I_t使用更大的输入窗口外，对于先前和将来的信息，我们可以考虑使用非对称链，也可以考虑使用关于参数和（或）与后验学习方法联系在一起的构架的先验知识。我

⑩ SSpro可通过http://promoter.ics.uci.edu/BRNN-PRED访问。

们还可以用一种带有不同记忆能力的多个“轮”组成的“多轮阵列”，沿蛋白质的不同方向滚动并且可能跨越更短的距离。这一方法可能优于两个“轮”的方法。值得注意的是，使用多层感知器实现$\beta(\cdot)$和$\phi(\cdot)$仅仅是一种选择。例如，递归径向基函数或2阶RNN的推广是容易实现的参数化方法。最后，本节中描述的思想可以应用于处理生物信息学的其他问题，也可以应用于适用非因果方法的其他领域。对于一般方法的进一步拓展，显然包括蛋白质功能特征的预测，如信号肽的预测。



第10章　进化的概率模型：系统进化树

10.1　进化的概率模型简介

这一章主要讨论生物进化以及如何由序列数据推断系统进化树（phylogenetic tree）。之所以把这部分内容包含进来，一方面是因为序列进化是计算分子生物学的一个中心问题，另一方面是因为这里用到的思想和算法能够很好地再次阐释第2章中讲述的广义概率推断体系。

自达尔文时代以来，生物（尚生存的或已灭绝的）之间的进化关系一直通过形态和（或）生化特性来推测。如今，人们普遍采用DNA和蛋白质序列来得到系统进化树。[182]由于DNA分子极其稳定，人们甚至可以从灭绝了多年的生物体内提取出大段完好的DNA。[251]利用其DNA，人们已经把早已灭绝了的与象相近的猛犸象定位到系统进化树上；对于死去的人，人们可以据此建立他们的准确家系关系。在最近的研究中，人们证实了俄国的末代沙皇尼古拉斯二世的身份，[211,274]还证明了安娜·安德森声称自己是沙皇遗失的女儿阿纳斯塔西娅的故事是假的。[212,477]沙皇的遗骨（和DNA）自1918年以来就一直被埋在土里。

文献中有很多从序列数据推断系统进化树的方法。大多数方法是以下两种主要方法的变形：吝啬法[181]和似然法。[178,519,269]显然，似然法以进化过程的概率模型为基础（见参考文献［295］）。实际上，“似然法”（likelihood metbod）这一名词常常被用在与一类特定的概率模型相联系的领域中。尽管吝啬法以进化模型形式独立描述，但实际上可以看做似然法的近似。

从广义贝叶斯体系和考克斯—杰恩斯公理出发，我们知道：为了从一组序列

中推断出系统进化树，必须先有一个进化的概率模型。最大似然估计（ML）法是我们根据这样一个模型所能够做的最基本的一步推断。目前文献中的所有其他方法都以ML为基础，包括参考文献［178］中基于某一类特定模型的ML法。正如我们已经看到的，HMM并非描述进化过程的一个完备模型。在分子水平上，进化不仅可以通过插入和缺失进行，而且也可以通过替换、倒置和转位来进行，因而必须采用不同的模型。下面首先介绍一些有关树的基本知识和概念。

10.1.1 树

一棵树T是一个无环连通图（connected acyclic graph）。在树里，每2个点由惟一的1条路径相连，而且顶点数总是严格地比边数大1。如果一棵树的每个顶点只有1个或3个邻居，那么这棵树就是二叉（binary）树。如果有一个节点r被选中作为根，那么这棵树就是有根（rooted）树。在系统进化树里，根用于表示祖先序列，所有其他序列都由这个序列演化而来。系统进化树（不论是有根树还是无根树）的两个重要特征是拓扑和枝长。拓扑指树随时间发生的分支模式。枝长经常以某种方式被用来表示事件之间的时间距离（图10-1）。

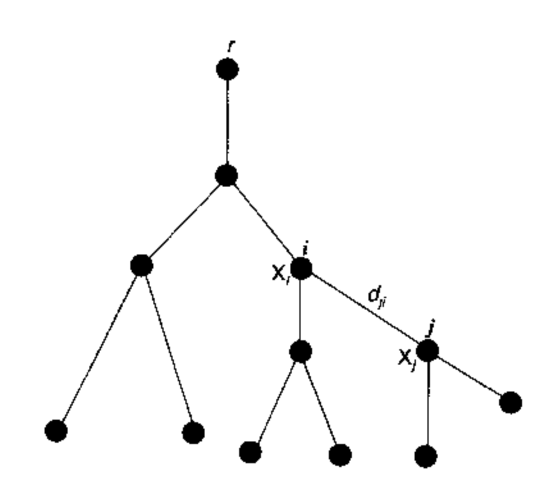


图10-1 简单的二进制系统进化树

r为根；d_{ji}是节点i和j间的时间距离；X_i为对应隐节点i的字符。底部的叶节点对应于观察到的字符。从i节点到j节点的替换概率为$p_{X_jX_i}(d_{ji})$。

10.1.2 概率模型

最基本却十分有用的概率进化模型也是简单骰子模型的一种变体。可以想像，

从一个祖先序列出发，进化是随机进行的，只存在替换，并且这种替换在各个位置间是独立的。如果我们考虑序列的一个给定位置i，且用$\chi^i(t)$表示时刻t位置i上的字符，我们可以做一个一般的马尔可夫过程假设，即对于$t>0$，概率

$$p^i_{\mathrm{YX}}(t)=\mathrm{P}(\chi^i(t+s)=\mathrm{Y}|\chi^i(s)=\mathrm{X}) \tag{10.1}$$

与$s(s\geqslant 0)$独立。因此，对每个位置i，存在一个概率$p^i_{\mathrm{YX}}(t)$，表示X经过t时间长的进化过程被Y替换的概率。因而这相当于在每个时刻t和每个位置i上有一个$|A|$面的骰子。为了进一步简化这个模型，现在做进一步的近似，即假设替换概率在所有位置上都是相同的，这样就有$p^i_{\mathrm{YX}}(t)=p_{\mathrm{YX}}(t)$。显然，对于任意X、Y和$t$，我们必有$p_{\mathrm{YX}}(t)\geqslant 0$且$\sum_{\mathrm{Y}} p_{\mathrm{YX}}(t)=1$。考虑到$t$时刻和$s$时刻事件的独立性，从（10.1）可以推出Chapman-Kolmogorov方程：

$$p_{\mathrm{YX}}(t+s)=\sum_{\mathrm{Z}\in A} p_{\mathrm{YZ}}(t)p_{\mathrm{ZX}}(s) \tag{10.2}$$

10.2　替换概率和进化速率

为了确定整个模型，剩下的所有工作就是确定替换概率$p_{\mathrm{YX}}(t)$。这与替换矩阵（如在第1章中我们曾经讨论过的PAM矩阵）有关。我们可以进一步合理地假设

$$\lim_{t\to 0^+} p_{\mathrm{YX}}(t)=\delta(\mathrm{Y},\mathrm{X})=\begin{cases}1 & Y=X\\ 0 & Y\neq X\end{cases} \tag{10.3}$$

如果我们用$P(t)$表示矩阵$P(t)=(p_{\mathrm{YX}}(t))$，则由（10.3）可定义$P(0)=Id$，其中$Id$是$|A|\times|A|$单位阵。可以证明$P(t)$的每一个矩阵元素都是可导的，于是记$P'(t)=(p'_{\mathrm{YX}}(t))$。整个进化模型完全可由它在$t$=0处的右导数确定：

$$Q=P'(0)=\lim_{t\to 0^+}\frac{P(t)-Id}{t} \tag{10.4}$$

由（10.4）可知

$$P'(\mathrm{t})=QP(t)=P(t)Q \tag{10.5}$$

其推导过程如下：$P(t+dt)=P(t)P(dt)=P(t)(P(0)+Qdt)=P(t)(Id+Qdt)$，其中第一个等式由（10.2）得到，第二个等式由（10.4）得到，于是有$P(t+dt)-P(t)=P(t)Qdt$。

如果令$Q=(q_{YX})$，根据（10.5），最终解由下式给出：

$$P(t)=e^{Q(t)}=Id+\sum_{n=1}^{\infty}\frac{Q^n t^n}{n!} \qquad (10.6)$$

注意：如果Q是对称的，则P也是对称的，反之亦然。这样的假设可以简化计算，但在生物上它并不现实，因此我们将不使用该假设。如果对于任何时刻t，都有$P(t)p=p$，则称一个分布列向量$p=(p_X)$是稳态的（stationary）。因此，一旦进入稳态分布，该随机过程就永远停留在该分布上。根据（10.4），这意味着p是Q的核，即$Q_p=0$。（10.6）也说明了这一点。如果我们假设被观察的序列是由系统在其稳态分布下产生的，那么p就可以很容易地由被观察到的序列的平均组成估计出来。

简要概括一下，我们定义了序列进化的一类概率模型。这类模型由四个假设来刻画：

1. 在每个位点，进化操作仅仅通过替换来完成，因而没有插入和缺失。所有观测序列必须具有相同长度；
2. 在每个位置上发生的替换与其他位置上发生的替换相互独立；
3. 替换概率仅仅依赖于当前状态，与历史状态无关（马尔可夫性质）；
4. 每个位置具有相同的马尔可夫过程。

真实的DNA进化不满足上述任何一个假设。在真实的DNA进化中，序列长度可以由于插入或缺失而改变；不同位置的进化不是独立的；进化速率不论是在时间上还是在位置函数上都不是均匀的；真实的DNA存在重组现象。尽管如此，上面的假设仍是一种有用的初步近似。当前很多研究集中在如何放宽这些假设上，前两个假设可能最难以放宽。对于插入和缺失，人们可以在现有体系的字符集A中加入一个间隙符，虽然这种做法并不让人完全满意。无论如何，为了在上述模型类中进一步确定模型，必须提供速率矩阵Q。

10.3 进化速率

值得注意的是速率矩阵Q相当于一个乘子，因为对于任意$\lambda\neq 0$，有$P(t)=\exp(Qt)=\exp[(\lambda Q)(t/\lambda)]$。在模型类的一个简单的子类里，假设$\lambda dt$是给定位置上一小段时间内替换发生的概率。于是，$\lambda$是单位时间的替换率。进一步地，如果发生了替换，字符将以概率$p=(p_X)$被选中。于是我们有

$$p_{YX}(dt)=(1-\lambda dt)\delta(Y,X)+\lambda dt p_Y \qquad (10.7)$$

对于任意X、Y，这等价于按照下式确定Q：

$$q_{XX}=\lambda(p_X-1) \quad 和 \quad q_{YX}=\lambda p_Y \tag{10.8}$$

注意到$e^{-\lambda t}$是一个时间段t内不发生任何替换的概率，由（10.8）和（10.6），或直接由（10.7）可以得到：

$$p_{YX}(t)=e^{-\lambda t}\delta(Y, X)+(1-e^{-\lambda t})p_Y \tag{10.9}$$

（10.7）中的分布p可以任意选择，这一点很有用。但是，一旦被选定，可以证明它是（10.9）的稳态分布，它就有了与上面所述的性质。如上所述，如果假设数据处于平衡态，p可以由数据直接得到。

另外，$p_{YX}(t)$只通过乘积λt与t相关。在缺少任何其他条件的情况下，可以选$\lambda=1$，这时是以替换发生的期望数目为单位来度量时间t的。如果允许λ沿着树的每个枝改变，就等价于在不同的枝上采用了不同速率的时钟来测量时间。于是沿所有可能的路径，从根到叶节点的总长度可以不再是一个常数。

由（10.9）定义的过程的另一个有用特性是可逆性，即替换过程沿着时间轴向前和向后看都是一致的。考虑到如下事实，这一点就不难理解了：由（10.9）可以得到平衡方程

$$p_{YX}(t)p_X=p_{XY}(t)p_Y \tag{10.10}$$

而其他的概率进化模型也满足可逆性。[302]

10.4　数据似然度

给定一组序列和概率进化模型，可以试着寻找最可能的树的拓扑结构和最可能的枝的长度。[178,519]这就是使用“系统进化的ML法”这种表述方式的原因所在。

我们首先假设在字符集A上有K个序列，所有序列长度都为N，相应地给出一棵系统进化树T，具有根节点r，相邻顶点i和j之间的时间长度为d_{ji}。第一个目标是根据前文所述的进化的马尔可夫模型计算似然度$\mathbf{P}(O_1, \cdots, O_K|T)$。根据各列之间的独立性假设，有

$$\mathbf{P}(O_1,\cdots,O_K|T)=\prod_{k=1}^{N}\mathbf{P}(O_1^k,\cdots,O_K^k|T) \tag{10.11}$$

其中，O_j^k表示第j个序列的第k个字符。这样，只需研究与第k列对应的$\mathbf{P}(O_1^k,\cdots,O_K^k|T)$

这一项，它在树的K个叶节点上具有字符O_j^k。在以下的讨论中，将用一般性的记号O来表示在一个固定位置上观察到的字符集。我们可以认为，在树的每个顶点i上有一个隐随机变量χ_i，表示顶点i上的字符。于是，这样一棵系统进化树就可以看成是一个简单的贝叶斯网络（见附录C），该网络具有树的结构且给定父节点i，节点j的条件概率具有参数d_{ji}（时间距离），其形式为

$$\mathbf{P}(\chi_j=\mathrm{Y}|\chi_i=\mathrm{X})=p_{\mathrm{YX}}(d_{ji}) \qquad (10.12)$$

因此，所有常用的有关贝叶斯网络的算法都可以用于这一简单情形。特别地，似然度$\mathbf{P}(O|T)=\mathbf{P}(O_1^k,\cdots,O_K^k|T)$可以通过两种途径来计算：一种从根节点开始算，一种从叶节点开始算。

如果从根节点开始计算，用（X_i）表示分配给内部节点I（包括根节点r但不包括叶节点）的字符。分配给内部节点的字符扮演的当然是隐变量的角色，类似于第7章的HMM路径。在这里，记X_i为分配给顶点i的字符，该记号可以加以扩展，使之包含那些在叶节点上观察到的字符。这样一种全局分配的概率很容易被计算出来：

$$\mathbf{P}(O,(\mathrm{X}_i)|T)=\mathbf{P}((\mathrm{X}_i)|T)=p_r(\mathrm{X}_r)\prod_{i\in I}\prod_{j\in N^+(i)}p_{\mathrm{Y}_j\mathrm{X}_i}(d_{ji}) \qquad (10.13)$$

其中，p_r是根节点字符的先验分布。$N^+(i)$表示顶点i的子节点集合，边的方向是从根节点指向叶节点。假设过程处于平衡状态，p_r是稳态分布$p=p_r$并可以通过平均组成加以估计。观测似然度通过对所有可能的分配求和来计算：

$$\mathbf{P}(O|T)=\sum_{(\mathrm{X}_i)}p_r(\mathrm{X}_r)\prod_{I-\{r\}}\prod_{j\in N^+(i)}p_{\mathrm{Y}_j\mathrm{X}_i}(d_{ji}) \qquad (10.14)$$

以上求和式含有$|A|^{|T|-K}$项，因此计算效率很低。其中$|T|$是树的数目。

递归地将数据信息从被观察的叶节点向根节点传播，能够大大提高似然度计算的效率。用$O^+(i)$表示以顶点i为根节点的子树所包含的数据信息，即在i的后代叶节点上观察到的字符，则若i是树的叶节点，有

$$\mathbf{P}(O^+(i)|\chi_i=\mathrm{X},T)=\begin{cases}1 & \text{若X在}i\text{处观察到}\\ 0 & \text{若X未在}i\text{处观察到}\end{cases} \qquad (10.15)$$

如果叶节点上是什么字符比较模糊，那么可以利用另一种分布。如果i是任意内部节点，则

$$\mathbf{P}(O^{+}(i)|\chi_i=\mathsf{X},T)=\sum_{\mathsf{Y}\in A}\sum_{j\in N^{+}(i)}p_{\mathsf{YX}}(d_{ji})\mathbf{P}(O^{+}(j)|\chi_j=\mathsf{Y},T) \tag{10.16}$$

数据信息O可以以这种方式传播到根节点r。这样，容易得到如下完整的似然度：

$$\mathbf{P}(O|T)=\sum_{\mathsf{X}\in A}p_r(\mathsf{X})\mathbf{P}(O^{+}(r)|\chi_r=\mathsf{X},T)=\sum_{\mathsf{X}\in A}p_r(\mathsf{X})\mathbf{P}(O|\chi_r=\mathsf{X},T) \tag{10.17}$$

这一算法仍旧是一种贝叶斯网络的传播算法，有时称之为“剥皮”（peeling）算法或“剪枝”（pruning）算法。注意：对于每一列，均可选择不同的p_r平均组成和$p^k_{\mathsf{YX}}(d_{ji})$概率，但前面的计算结构不改变。因此，每个位点的进化模型相似，但不必完全相同。另外值得注意的是，还可以计算出内部节点的一个最优（最有可能的）分布，而不是去综合内部节点的所有可能分布。这等价于我们在HMM中的Viterbi路径计算。

一个有用的结论是，如果进化模型是可逆的，且对根节点的位置没有任何外部约束（例如要求所有的叶节点都是同时期的），那么似然度与根节点的位置相互独立。前向过程和后向过程相等，根可以沿着树的边任意移动，从而在整棵树内任意移动。更进一步地，考虑一棵树，它有根节点r，两个子节点i和j，在从r到j的枝上有一个备选根节点s（图10-2）。从（10.16）和（10.17）我们有

$$\mathbf{P}(O|T)=\sum_{\mathsf{X},\mathsf{Y},\mathsf{Z}\in A}p_r(\mathsf{X})p_{\mathsf{YX}}(d_{ir})\mathbf{P}(O^{+}(i)|\chi_i=\mathsf{Y},T)p_{\mathsf{ZX}}(d_{sr})\mathbf{P}(O^{+}(s)|\chi_s=\mathsf{Z},T) \tag{10.18}$$

考虑可逆性并假设系统处于平衡状态：$p=p_r=p_s$且$p_r(\mathsf{X})p_{\mathsf{ZX}}(d_{sr})=p_s(\mathsf{Z})p_{\mathsf{XZ}}(d_{rs})$。现在有

$$\sum_{\mathsf{Y}\in A}p_{\mathsf{YX}}(d_{ir})\mathbf{P}(O^{+}(i)|\chi_i=\mathsf{Y},T)=\mathbf{P}(O^{++}(r)|\chi_r=\mathsf{X},T) \tag{10.19}$$

其中“++”表示以s而非r为根节点的树的消息。类似地

$$\mathbf{P}(O^{+}(s)|\chi_s=\mathsf{Z},T)=\sum_{\mathsf{W}\in A}\mathbf{P}(O^{++}(j)|\chi_j=\mathsf{W},T)p_{\mathsf{WZ}}(d_{js}) \tag{10.20}$$

集中上面各式，最终有

$$\mathbf{P}(O|T)=\sum_{\mathsf{X}\in A}p_s(\mathsf{X})\mathbf{P}(O^{++}(s)|\chi_s=\mathsf{X},T) \tag{10.21}$$

因此，可以自由地把根节点放到树的任何位置上却不改变似然度，对与无根树相关联的等价类别，我们也可以讨论似然度。

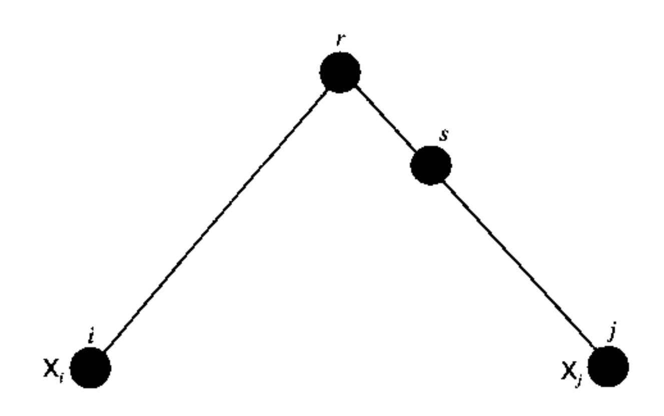


图10-2 以r为根节点的树，在从r到j的枝上有一个备选根节点s

10.5 进化树的优化和学习算法

迄今为止，很少有工作涉及如何根据分枝过程和分枝长度来定义系统进化树空间的先验分布。因此，我们可忽略先验分布问题而直接进入贝叶斯推断的第一步：估计ML树。对一棵给定了拓扑结构和枝长的树，10.4节中计算了它的似然度。如果拓扑结构给定，长度d_{ji}可以看成模型的参数，能用ML法优化。与HMM一样，一般地说，ML估计不能解析地给出，但可以用梯度下降法、EM或Viterbi学习算法的某种形式近似求得。作为练习，读者可以自己推导出用EM或梯度下降法优化枝长的方程。[178]

10.5.1 最优拓扑

拓扑的优化是第二个问题，这需要进行近似处理。由于可能存在的树，甚至无根树的数目都是指数级的，搜索无法穷尽整个拓扑结构空间。参考文献［178］描述了一种启发式算法，用于在这个空间中搜索较好的拓扑结构，这里就不详细回顾了。一个广泛采用的启发式算法是从一棵只有两个物种的树开始，一个一个逐步地加入新的物种（即观测序列）。在每一步中选出一个新物种，考虑它在当前的树中所有可能的位置。选出那个最有可能的位置，然后进行下一步。需要注意的是，用这类搜索算法，树的最终拓扑结构依赖于观测序列的表示顺序。

无论如何，用ML法解决系统进化树的问题，在计算上显然是十分复杂的。对系统进化的完整贝叶斯处理更加复杂，因为除了借助先验分布，还需要通过对这些树依次积分来估计一个给定的替换在过去是否发生的概率。吝啬法可以看做ML法的快速近似模型。

10.6 吝啬法

吝啬法的基本思想是：最优树是这样一棵树，沿着它的分枝发生替换的次数最少。在这个意义上，它与MDL（minimum description length，最小描述长度）思想有些联系。更公式化的表述是，再次考虑树的内节点的一个分配(X_i)，并把此记法扩展到叶节点，叶节点上的字符是固定的并由观测确定，由此定义分配的吝啬代价（parsimony cost）为

$$\mathcal{E}_p((X_i)|T)=\sum_{i\in I}\sum_{j\in N^+(i)}\delta(X_i, X_j) \quad (10.22)$$

换句话说，对每个非恒等替换都引入了一个固定代价，而我们的目标是找到一种分配使树的代价最小。对于给定的树，最小分配也称做最小突变拟合（minimum mutation fit）。

为了考察吝啬法与ML法的联系，回顾对于给定的树，有一种分配(X_i)的概率为

$$\mathbf{P}((X_i)|T)=p_r(X_r)\prod_{i\in I}\prod_{j\in N^+(i)}p_{X_jX_i}(d_{ji}) \quad (10.23)$$

其负对数概率是

$$\mathcal{E}((X_i)|T)=-\log p_r(X_r)-\sum_{i\in I}\sum_{j\in N^+(i)}\log p_{X_jX_i}(d_{ji}) \quad (10.24)$$

如果令

$$\mathbf{P}_{X_jX_i}(d_{ji})=\begin{cases}a & 若X_j=X_i\\ (1-a)/(|A|-1) & 若X_j\neq X_i\end{cases} \quad (10.25)$$

其中$1/|A|<a<1$，则容易验证存在两个常数$\alpha>0$和β使得

$$\mathcal{E}=\alpha\mathcal{E}^P+\beta \quad (10.26)$$

事实上，$\alpha=\log[a(|A|-1)/(1-a)]$，$\beta=-|E|\log a+\log|A|$，其中$|E|$为树$T$的边数。换句话说，给定树的最小突变拟合等价于由（10.25）定义的同一拓扑树的ML系统进化模型中给出的Viterbi（最有可能的）分配。因此吝啬法可以看做是系统进化的ML法的一种近似。它有一个隐含的假设，即变化较少且在字符集和时间上是均匀的。因此，如果变化的总数在所考虑的进化时段中很小，吝啬法在统计上是合理的。吝啬法的递归算法为人熟知，见参考文献［181］。在加权吝啬法[464]中，我们

可以通过对每种替换引入不同的权重$w(\mathrm{Y}, \mathrm{X})$，放松替换在字符集上必须均匀的假设。显然，这可以视为在一种特殊的ML环境下，对A中的任意Y使得

$$p_{\mathrm{YX}}(d_{ji})=\frac{e^{-\alpha w(\mathrm{Y},\mathrm{X})}}{\sum_{\mathrm{Z}\in A}e^{-\alpha w(\mathrm{Z},\mathrm{X})}} \tag{10.27}$$

吝啬法在计算上比ML法快，这大概是它被广泛应用的一个原因。然而，当进化很快时，吝啬法可能导致错误。可以用进化的概率模型产生人工数据以比较ML法和吝啬法。对于小样本，ML法和吝啬法显然都可能导致错误的系统进化结论。然而对于大样本，ML法经常能够构造出正确的系统进化树，而吝啬法却不总是如此。

10.7 扩 展

以上我们回顾了构造系统进化树的基本方法。值得注意的是，构造系统进化树是贝叶斯推断方法的又一应用。构造系统进化树的第一步要建立可操作的进化过程概率模型。马尔可夫替换模型构成了这样一类模型。目前主要采用的系统进化树的构造方法（包括吝啬法），都是在这样一类模型里进行ML推断的特例或近似。系统进化树的构造算法在计算上很困难，特别是在需要研究大量可能的树的情形下。

进化的HMM和概率树模型有一些互补的优缺点。由HMM产生多重序列比对是进化重构算法的起点。需要利用进化模型正则化HMM，即通过处理多重序列比对中每列的原始计数得到生成概率，将其做细微调整用于大规模数据库的同源性搜索。很明显，有一个研究方向把它们结合起来，即交叉树、序列比对、系统进化以及结构，[247,519]以及提出允许插入和缺失的进化过程的概率模型，并保证它们在计算上的可操作性（参见附录C的树结构HMM）。

单一的马尔可夫替换过程并不是一个好的进化模型，其原因不仅包括在本章里讨论过的那些，而且还由于模型在很长的进化历史中只考虑了一次平衡分布。这与观测是不一致的，而且与用Dirichlet混合分布（参见附录D）作为HMM生成概率的先验分布等也不一致。经过模型的驰豫时间（relaxation time）以后，简单的马尔可夫模型不能产生分布的聚类和Dirichlet混合分布的不同分量。为了解释可能的聚类，必须利用这个简单模型跨越一个较短的转变期，或转移到更高层次的进化模型。

高层次的进化模型应该是什么样子呢？想像我们可以观察到在进化过程的不

同时段（如每1亿年）产生的多重序列比对。在每个观测时段，比对的列表示从可能列的复杂分布中抽取的一个样本。正是这一分布在随着时间进化，因此在这类更高层次的模型中，进化在分布不断演化所构成的更高层次的分布上发生。这类模型的一个简单例子可以按如下方法构造。可以想像原始生成分布（$t=0$时）是一个Dirichlet混合分布$\mathbf{P}(P)=\sum_i \lambda_i^t \mathcal{D}_{\alpha_i^t Q_i^t}(P)$，而且存在一个简单的在$Q$上操作的马尔可夫替换过程（可能还有在$\alpha$和$\lambda$上操作的附加过程）。在$t$时刻，这个分布变成$\mathbf{P}(P)=\sum_i \lambda_i^t \mathcal{D}_{\alpha_i^t Q_i^t}(P)$。例如使用PAM矩阵替换模型，如果$Q_i$等于仅在位置$i$上取值为1（表示字符X）的二值单位向量，那么在时刻t的Q_i^t与相应的PAM矩阵的第i列相对应［表示$p_{\mathrm{X}}(t)$］。这样一个模型与经过Dirichlet混合分布正则化的HMM生成概率相一致，[497]其中Dirichlet混合分布与PAM矩阵中的列向量有关。



第11章　随机文法和语言学

11.1　形式文法的介绍

本章讲述最后一种序列的概率模型：随机文法（stochastic grammar）。随机文法的基本思想是第3章的简单骰子模型和HMM的直接扩展。

第1章简要提到过，形式文法（formal grammar）最初是为了建立自然语言模型而发展起来的，大约在同一时期，沃森和克里克发现了DNA的双螺旋结构。从那时起，文法就被广泛应用于分析和设计计算机语言和编译器。[3] 文法是对字符串建模的很自然的工具，最近也被应用到生物序列研究中。事实上，计算分子生物学中的很多问题可以用形式文法研究。[91,479] 这里，随机文法的基本目标是通过机器学习的方法找到数据对应的文法。

本章将回顾形式文法的基本理论，包括几类不同的文法、它们的特性、乔姆斯基层次（Chomsky hierarchy）及其同HMM的联系。在11.3节中，我们将示范随机文法是如何应用到生物序列中的，特别如何针对RNA分子应用上下文无关文法（context-free grammar）。接下来的三节集中讲述先验、似然度及学习算法。最后两节讲述它们的主要应用。

11.2　形式文法和乔姆斯基层次

11.2.1　形式语言

先从字符集A开始。A上所有长度有限的字符串构成的集合表示为A^*。$\varnothing$表示空

串。一种语言是A^*的子集。在某种意义上，可以说是序列中的启动子和接纳体位点构成了定义在DNA字符集上的一种语言。这个定义本身用处不大，除非我们能确定语言的生成、识别和分类的简单规则。文法可以看做生成一种语言的规则的集合。

11.2.2 形式文法

可以生成并且只能生成所有句法正确的串，其所遵循的规则集合称为形式文法。一个形式文法G包含：一个字符集A，称为终结符；变量的字符集V，其中的变量也称为非终结符；以及一个产生式规则（production rule）组成的集合R。在非终结符中有一个特殊的变量s表示开始变量。每个产生式规则包含一对（α, β），常常记为$\alpha \to \beta$，其中α和β是$(A \cup V)^*$中的元素。$\alpha \to \beta$中的箭头可以读做“产生”或“扩展为”。我们用希腊字母表示由非终结符和终结符组成的串。因此在通常意义上，α和β是由字母和变量组成的串。另外，设定α至少包含一个非终结符。给定G和（$A \cup V$）上的两个串γ和δ，如果存在串的有限序列$\pi=\alpha_1,\cdots,\alpha_n$，使得$\gamma \to \alpha_1 \to \cdots \to \alpha_n \to \delta$（也写做$\gamma \to_{\pi} \delta$），其中的每一步都对应$R$中的产生式规则的一次应用，那么就称$\delta$可以从$\gamma$导出。由文法$G$生成的语言$L=L(G)$是可以从开始状态$s$导出的全部终结符串的集合。

例如，考虑由$A=\{X, Y\}$，$V=\{s\}$，$R=\{s \to XsX, s \to YsY, s \to X, s \to Y, s \to \varnothing\}$定义的文法。字符串XYYX可以由初始的串$s$如下导出：$s \to XsX \to XYsYX \to XYYX$，依次应用第一、第二和第四条产生式规则。更一般地，容易证明G生成A上的全部回文（palindrome）的集合。回文是向前读或向后读都相同的字符串。现在我们可以定义几种不同类型的文法和乔姆斯基层次。乔姆斯基层次就是按照复杂性和表达能力的级别对文法进行的分类。

11.2.3 乔姆斯基层次

表11-1总结了乔姆斯基层次和它的特性。

表11-1 文法、产生式规则及其等价关系一览表

	正则文法	上下文无关文法	上下文相关文法	递归可枚举文法
产生式规则	$u \to Xv$ $u \to X$	$u \to vw$ $u \to X$	$\alpha X\gamma \to \alpha\beta\gamma$	所有
闭包性质	$\cup, \cdot, ^*$ $\cap, ^-$	$\cup, \cdot, ^*$ 无$\cap$，无$^-$	$\cup, \cdot, ^*$	所有
等价的自动机	有限状态自动机	下推自动机	线性有界的图灵机	图灵机
特征语言		回文	复制语言	所有
相关性特征	无长程相关性	嵌套	交叉	所有

正则文法

最简单的一类文法是正则文法（regular grammar, RG）。在正则文法中，一条产生式规则的左边是单个变量，右边则是字符集中的单个字符后最多再加上一个变量。因此串只能在一个方向上增长。准确地说，如果所有的产生式规则是$u \to Xv$、$u \to X$或$u \to \varnothing$的形式，则文法G是正则的（或右线性），其中u和v是单个的非终结符。如果一种语言可以由正则文法生成，则称它是正则语言。正则语言也可以用别的方式描述，如正则表达式就可以很有效地识别正则语言，尽管它的表达能力有限。

上下文无关文法

正则文法是上下文无关文法（context-free grammar，CFG）的特例。上下文无关的意思是使用表达式替换变量时，并不依赖于被替换的变量的上下文。准确地说，如果R中的所有产生式规则都是$u \to \beta$的形式（其中u是单个的非终结符），则文法G是上下文无关的。如果一种语言可以由上下文无关文法生成，则称其为上下文无关的语言。上下文无关文法可以用规范的形式表述，称为范式（normal form），例如乔姆斯基范式或格雷巴赫（Greibach）范式。如果一种上下文无关文法的每个产生式规则都是以下三种形式之一：（1）$s \to \varnothing$，（2）$u \to vw$，其中u、v、w是非终结符，（3）$u \to X$，则称其符合乔姆斯基范式。另外，如果$s \to \varnothing$在R中，则（2）中的v和w必须不同于s。

上面提到的回文文法是上下文无关的但不是正则文法。上下文无关文法常常用于定义计算机语言的句法以及构造编译器。可以想到，并非所有的语言都是上下文无关的。例如，复制语言（copy language）就不是上下文无关的。一个复制语言包括所有这样的字符串，其中后一半与前一半完全相同。XXYXXY就属于一个复制语言（对应于DNA中的直接重复）。尽管复制语言看起来与回文类似，但它们确实需要一类更复杂的文法。上下文无关文法也被用于对自然语言建模，但是由于自然语言不是上下文无关的，该文法取得的效果很有限。

上下文相关文法

在非上下文无关的文法中，我们可以定义上下文相关文法（context-sensitive grammar，CSG）的子类。如果文法中所有的产生式规则都是$\alpha X \gamma \to \alpha \beta \gamma$这样的形式，其中X在$A$中，[11] $\beta \neq \varnothing$（X可以在上下文α-γ中被替换为β），则称其为上下文相关文法。另外，如果s不出现在任何产生式规则的右边，则允许有规则$s \to \varnothing$。如果一种语言可以由上下文相关文法生成，则称它是上下文相关语言。可以证明

⑪ X应该是变量，它在V中而不是在A中。 ——译者注

复制文法是上下文相关的而不是上下文无关的。上下文相关文法的特点是产生式规则的右边至少要和左边一样长。

递归可枚举文法

这是范围最广的文法，没有上述的任何限制。递归可枚举（recursively enumerable）指的是：如果一个词属于这种语言，那么它的导出过程总是可以在有限的时间内通过图灵机获得，只要穷举所有可能（可数的）的推导就可以了。递归可枚举比递归弱：一般来说，无法在有限时间判定一个词是否属于一种语言，例如产生常见的停机问题。乔姆斯基层次指出以上几个主要文法类别构成严格递增的序列，也就是

$$RG \subset CFG \subset CSG \subset REG \tag{11.1}$$

所有的包含关系都是严格的，其中RG=正则文法，CFG=上下文无关文法，CSG=上下文相关文法，REG=递归可枚举文法。在乔姆斯基层次的较高层次允许有更通用的规则，但是也通过排除了更多的字符串而对语言有更多的限制。

11.2.4 二义性和语法分析

一个导出过程可以被排列成一个树形结构，称为分析树（parse tree），它反映了序列的句法结构。分析可以自上而下进行，也可以自下而上进行。如果序列拥有不止一棵分析树，则称它是二义性的。二义性（ambiguity）的概念对于编译器的设计很重要。二义性使得语法分析复杂化，它一方面使分析算法复杂化，另一方面可能使分析树数量随着被分析的字符串的长度呈指数增长。有一些用于分析特殊文法的算法和复杂性研究结果。如果所有产生式规则的右边至多包含一个非终结符，那么就称该文法是线性的。对于线性的上下文无关文法存在快速分析算法。一般说来，在乔姆斯基层次中，较高层语言的识别与序列文法分析需要更大的计算量。

11.2.5 闭包性质

乔姆斯基层次中的每个文法类对于很多语言操作都是封闭的或者说稳定的，例如“并”($L_1 \cup L_2$)，“串联”($L_1.L_2$)，“重复”(L_1^*)。正则文法对“补”($\bar{L}_1$)和“交”($L_1 \cap L_2$)也是封闭的。而上下文无关文法对“补”和“交”不是封闭的。

11.2.6 相关性

还可以从生成模式和自动机这两个角度考察文法。正则文法可以有整体的模式，例如形成XYXYXYXY这样的交替字符串。像HMM一样，正则文法不能处理

字符串中的长程相关性。上下文无关文法可以对一定的简单长程相关性建模，例如嵌套相关性（nested dependency）。如果所表示相关性没有相互交叉，那么它就是嵌套的。嵌套相关性是上下文无关语言的特征，例如回文，其中第一个字符必须和最后一个匹配，第二个必须和倒数第二个匹配，等等。如果相关性有交叉，例如复制语言，那么就需要用到上下文相关语言，因为在上下文相关语言的推导中必须自由移动非终结符，只有这样才能实现有交叉的相关关系。

11.2.7 自动机

理解乔姆斯基层次的最后一种方法是考察与每种语言对应的自动机（automata）。对此这里不深入研究细节问题，只是给出结论。正则文法对应于有限状态自动机（finite state automata, FSA），通常每个状态对应文法中的一个非终结符，像HMM一样。在这种自动机中，除了状态自身外没有任何储存机制——所有的信息都必须“硬编码”。上下文无关文法对应于下推自动机（pushdown automata, PDA），它和有限状态自动机类似，但是有一个存储栈。在每个时刻，只有栈顶是可以访问的。这种单点存储的机器可以通过每次对一个字符进行进栈或出栈操作实现回文。这种自动机不能处理交叉相关性，因为它在每个时刻都只能访问栈顶。上下文相关语言对应于可移动域线性有界的图灵机，其可移动域长度与输入/输出串的长度成正比。需要在移动区域上左右移动，以便复制和处理交叉相关性。而最一般的递归可枚举语言对应于可移动域无边界的图灵机（Turing machine, TM），这也是通用计算机的标准模型。

11.2.8 随机文法和HMM

到现在为止，我们考虑的都是确定文法。随机文法是通过在产生式规则上添加概率结构得到的。每个产生式规则$\alpha \to \beta$都被赋予一个概率$\mathbf{P}(\alpha \to \beta)$，并使得$\sum_{\beta} \mathbf{P}(\alpha \to \beta)=1$。因此，随机文法最大的特征是可以被看做是其所对应的语言（亦即由底层隐含的确定文法生成的语言）的一个具有一套参数w的概率生成模型。

现在应该可以清楚地看出，HMM可以看做一个随机正则文法（stochastic regular grammar, SRG）。为了说明这一点，只要把HMM中由字符X生成的从状态s_j到s_i的转换用SRG中概率为$t_{ij}e_{iX}$的产生式规则$s_j \to \mathrm{X}s_i$替换就可以了。这样，随机上下文无关文法（stochastic context-free grammar, SCFG）构成了更一般的一类模型。在下面的章节中，它们将被用于对RNA序列的结构建模，也可以看做第3章中的骰子模型的进一步推广。这种SCFG包含了一种每面两个字符的骰子。在最简单的RNA模型中，这两个字符反映了碱基的互补性质。因而，RNA骰子有四个

面，类似一个简单的DNA骰子，但四个面上的字符是AU、UA、CG和GC（不包括GU和UG）（参见图11-1）。



图11-1 DNA中沃森—克里克碱基对互补性的图示

在RNA中，尿嘧啶（U）代替了胸腺嘧啶（T）。

11.2.9 图文法

以上已经研究了普通字符集上的文法。我们还可以研究更一般的字符集，其中的“字符”是图或者图像处理中的像素图形。在图文法（graph grammar）中（见参考文献［165,158］及其列出的其他论文），人们必须详细确定，在推导过程中图是如何相互连接起来的。图文法有相当强的表达能力，也是生物大分子的二级结构和三级结构建模的自然选择。但是迄今为止，在这个方面还没做什么工作。一个关键的问题是对一般的图文法（甚至对特定类型的图文法）尚缺少有效的学习算法。

11.3 文法在生物序列中的应用

人们希望最终能在基因、染色体甚至基因组的尺度上建立文法模型。毕竟，在所有可能的长度相近的DNA序列中，基因组只占了很小的一部分。但是作为开始，必须考虑包含较少文法的较简单的例子，例如RNA二级结构和回文。

11.3.1 RNA二级结构和生物学回文

RNA二级结构

生物大分子的很多成分由RNA构成。重要的几类RNA包括转移RNA(tRNA)、核糖体RNA(rRNA)、剪接体中的小分子细胞核RNA(snRNA)、信使RNA(mRNA)以及各种类别的内含子。另外，还包括一些在试管中分离出的具有特殊功能（诸如蛋白质结合和催化功能）的小RNA分子家系。[109,356,469,55]

尽管RNA通常是单链的，但由互补碱基对形成的螺旋可以控制RNA的折叠，从而构成特殊的三维结构。从RNA链到功能分子的折叠过程主要由沃森—克里克碱基对A-U和G-C配对决定，在某种程度上也受G-U及更少见的G-A配对影响。RNA核苷酸的相互作用构成茎（stem）、环（loop）、凸起（bulge）等二级结构，而序列中相互远离的核苷酸在某些地方发生相互作用则形成伪结（pseudoknot）。[573] 这些配对常常有嵌套的结构，因此不能用正则语言或HMM有效建模。下面首先考虑RNA和其他分子中的生物学回文问题。

生物学回文

有很多RNA/DNA回文的例子，例如，蛋白质结合位点回文。生物学回文与前面描述过的回文略有差别，因为从两端开始配对的字符并不相同，而是互补的。例如AGAUUUCGAAAUCU是一个RNA回文。在DNA中这样的回文被称为反向重复。

由于DNA是双螺旋互补结构，螺旋中一股上的回文的每一半在另一股上都有镜像。因此，如果回文串在一股上从左到右读，那么在另一股上可以从右到左读到相同的字符串。RNA回文的长度是任意的，因此可能需要用上下文无关文法或更复杂的文法来建模（从技术上来说，有固定长度上限的回文可以用正则文法建模）。RNA回文通常折叠产生发卡（茎-环）结构（hairpin structure）。

形成RNA回文的一个语法如下：

$$s \to \mathrm{A}s\mathrm{U} \mid \mathrm{U}s\mathrm{A} \mid \mathrm{C}s\mathrm{G} \mid \mathrm{G}s\mathrm{C} \mid \varnothing \tag{11.2}$$

在一行中列出了所有可选的产生式规则，中间用“|”分开。可以按如下过程产生一个回文：$s \to \mathrm{A}s\mathrm{U} \to \mathrm{AG}s\mathrm{CU} \to \mathrm{AGU}s\mathrm{ACU}$，等等。生成的分析树能反映碱基的配对情况（参见图11-2）。实际的RNA回文并不是完美的，但个别不匹配的碱基对不会破坏二级结构。某些碱基对不匹配，例如UG，与其他的不配对相比还是可以被容忍的。为此也有必要引入概率。发卡结构的茎部有不配对的碱基凸起也是很常见的。RNA的柔韧性一般不足以在发卡结构的顶端形成180度的转弯。通常会有至少3到4个不配对碱基构成的环，有时这个环还会长得多。同样地，DNA回

文的两个半边也可能被隔开相当大的距离。所有这些特点都可以被包括到文法中，但是会使规则复杂化。

上面的文法能产生对应于单个回文的串。而DNA和RNA都含有大量的复合回文，即连续的回文和递归的回文。连续的回文就是两个或更多的回文一个接一个肩并肩地出现。递归的回文就是一个回文被嵌套在另一个回文内部。与递归回文对应的RNA二级结构是一个茎，在它的侧面凸起了另一个茎。出人意料的是，得到递归回文很简单：只需要增加一条产生式规则$s \rightarrow ss$。变量s的复制允许在一个已存在的回文中的任何地方生成一个新回文。对应的文法所产生的结构是有分支的茎，它是一种规范的二级结构。最著名的例子可能是tRNA的三叶草结构。还有其他许多由环和嵌套的茎组成结构的例子，尤其是在rRNA中。递归回文文法是上下文无关的，但与简单回文不同，它是有歧义的。反向重复UGAUCA-UGAUCA既可以分解成单个的发卡，也可以当成两个或更多的肩并肩的茎，其长度不必相同。可变的分析树对应于可变的二级结构。结构二义性使同一个RNA元件具有不同的角色，这是已有例子的。DNA语言学中的其他二义性例子包括重叠基因——在HIV病毒中，基因组的某些片段能编码一个以上的基因，因为它们使用了有歧义的起始位点和阅读框。

11.3.2 RNA的上下文无关文法

一般说来，一个用于RNA的SCFG需要如下类型的规则：

1. 配对生成规则，对于沃森—克里克配对，该规则为

$$u \rightarrow \mathrm{A}v\mathrm{U} | \mathrm{U}v\mathrm{A} | \mathrm{C}v\mathrm{G} | \mathrm{G}v\mathrm{C} \qquad (11.3)$$

对于比较罕见的配对（按稀有性排序）

$$u \rightarrow \mathrm{G}v\mathrm{U} | \mathrm{G}v\mathrm{A} \qquad (11.4)$$

2. 在左边生成单个字符（不配对碱基）

$$u \rightarrow \mathrm{A}v | \mathrm{C}v | \mathrm{G}v | \mathrm{U}v \qquad (11.5)$$

3. 在右边生成单个字符（不配对碱基）

$$u \rightarrow v\mathrm{A} | v\mathrm{C} | v\mathrm{G} | v\mathrm{U} \qquad (11.6)$$

4. 生成单个字符（不配对碱基）

$$u \rightarrow \mathrm{A} | \mathrm{C} | \mathrm{G} | \mathrm{U} \qquad (11.7)$$

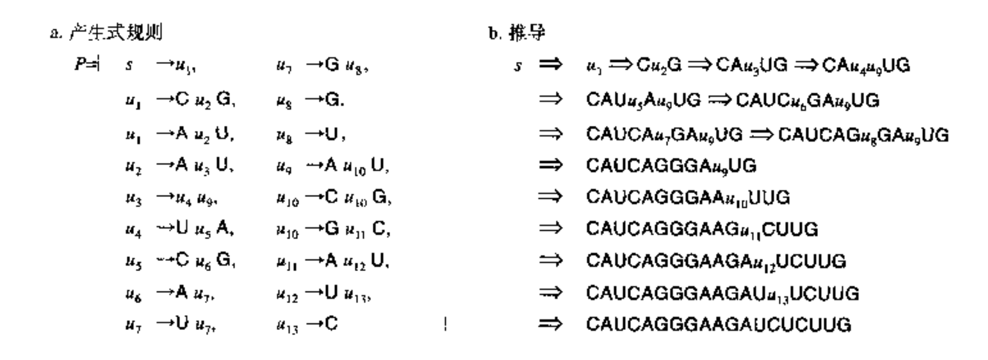
5. 分支（或分叉）

$$u \to vw \tag{11.8}$$

6. 删除（或跳过）

$$u \to v \tag{11.9}$$

产生式规则左边的非终结变量，例如u，扮演着HMM状态的角色，必须用u_1，u_2,…等编号。如同HMM，这些非终结变量可以分为三类：匹配、插入、删除或跳跃，每个都有不同的概率分布。匹配对应于RNA多重序列比对中重要的列。它与HMM的主要差别在于某些状态有生成两个配对符号的可能。对于插入状态对应的一个非终结符u，$u \to Xu$形式的产生式规则允许有多个插入，环区需要用它们来调整环的长度。图11-2显示了取自参考文献［460］的一个CFG RNA文法的例



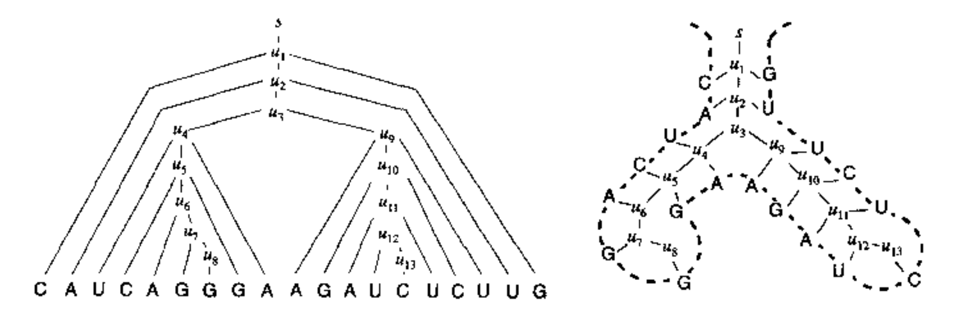


图11-2 简单的上下文无关文法及序列CAUCAGGGAAGAUCUCUUG的推导过程

a. 文法中的产生式规则集合，其中s是始始符号，u_1到u_{13}是非终结符。b. 推导过程。c. 推导过程对应的分析树。d. 反映了分析树结构的二级结构。摘自参考文献［460］。

子，图中同时显示了一个序列的文法生成过程、它的分析树以及二级结构。

当然上面列出的规则类型是冗余的，对RNA建模并不需要用到所有各种类型的规则的组合和所有类型的非终结符。参考文献［156］中的RNA协方差模型尽管名称不同，但其本质上与SCFG模型等价，其中只用到了下面的组合：

- 具有成对生成、左边单个生成和右边单个生成的匹配状态；
- 具有左边单个生成和右边单个生成的插入状态；
- 删除和分支状态。

当然，在文法的数量、训练所需时间以及是否对数据过拟合或欠拟合这几个方面，该模型需要做一些折中。

11.3.3 超越上下文无关文法

迄今为止我们还停留在上下文无关文法、下推自动机和嵌套相关性的领域内。很多简单的进化操作，例如插入、删除和替换都能用上下文无关文法单独表达。然而，在核酸字符串的区块上的其他遗传操作——例如重复、反向、易位及转座——产生了语言中的交叉相关性，因此不能用上下文无关文法正确表达。直接重复在DNA中相当普遍，其本质上构成了一种复制语言。据此，它们可以用上下文相关文法建模。相关性的交叉也可以在生物分子的二级结构和三级结构中见到。这里的一个例子是RNA结构中的伪结。

前面已提及，当一个单链环区和环外的一个互补序列构成沃森—克里克碱基对时出现伪结。伪结可以看成是交叉的而非嵌套的回文。例如AACCGGUU可以看成两个回文的嵌套：AAUU和CCGG。从另一方面看，AACCUUGG是一个伪结，因为这些互补的碱基对相互交叉。伪结这样的特征被归类为不规范的二级结构。前面的上下文无关文法不足以对伪结建模。伪结和直接重复一样，可以用上下文相关文法描述。最后必须提一下，DNA语言可以看成是其他几种语言的重叠或交叉，例如转录、剪接和翻译。即使每个单独的语言都是上下文无关的，但它们合并起来后不一定上下文无关。

11.4 先验信息和初始化

11.4.1 从多重序列比对学习文法规则及初始化

SCFG中的所有规则以及它们的概率，都可以很容易地从多重序列比对得到，做法和HMM一样，而且也会产生同样的问题。参考文献［156］给出了一种算法——产生式规则本身从一套未比对序列得到。对于大的RNA分子，构造文法的过程可

以按层分解。首先在二级结构的大尺度motif（超二元结构模体，如螺旋和环等）的基础上构造高层次的文法［在参考文献［460］中称为元文法（metagrammar）］，然后每个motif再各自用一套SCFG的规则表达。

11.4.2 Dirichlet先验分布

Dirichlet先验分布是随机文法的产生式规则的自然选择。在11.3.2节的列表里，主要需要考虑两种类型的规则：配对生成规则$u \rightarrow XvY$，以及环区的形式为$u \rightarrow Xv$的单个字符生成规则。对于RNA，第一类规则有16种可能的形式，第二类则有4种。由于沃森—克里克碱基配对关系，相应的Dirichlet向量不是单一的。它们可以很容易地从RNA结构比对数据库中得到，例如参考文献［346］（表11-2）。

其他规则，例如分支的产生规则，若有必要也可以Dirichlet正则化。

表11-2 在实际的观察频率中加入伪螺旋数以反映先验信息

		3' A	3' C	3' G	3' U
5'	A	0.160 097	0.135 167	0.192 695	1.590 683
5'	C	0.176 532	0.134 879	3.403 940	0.162 931
5'	G	0.219 045	1.718 997	0.246 768	0.533 199
5'	U	2.615 720	0.152 039	0.249 152	0.249 152

Dirichlet先验分布中的16个参数，是通过在16S个rRNA序列的大比对中统计配对碱基位置的分布得到的。从比对中也可得到环区中核苷酸分布的四参数的Dirichlet先验分布：A（0.26），C（0.21），G（0.18），U（0.20）。

11.5 似然度

首先考虑如何计算序列$O=X^1 \cdots X^t \cdots X^T$的似然度$\mathbf{P}(O|w)$，其中序列的文法$M=M(w)$，参数为$w$。考虑到SCFG是有歧义的，记$\pi=\alpha_1,\cdots,\alpha_n$表示从状态$s$开始的$O$的一个推导，则

$$\mathbf{P}(s \rightarrow_{\pi} O|w)=\mathbf{P}(s \rightarrow \alpha_1|w)\mathbf{P}(\alpha_1 \rightarrow \alpha_2|w)\cdots\mathbf{P}(\alpha_n \rightarrow O|w) \tag{11.10}$$

$$\mathbf{P}(O|w)=\sum_{\pi}\mathbf{P}(s \rightarrow_{\pi} O|w) \tag{11.11}$$

显然，这些表达式与HMM中得到的式子很类似，只是HMM路径被文法推导所

代替。另外，这些似然度的表达式并不能直接使用，因为可能的分析树的数目与序列的长度之间呈指数关系。但是，可以用动态规划方法绕过这个问题。对于非随机的乔姆斯基规范形式的上下文无关文法，这个算法被称为Cocke-Kasami-Younger算法。[393]与HMM的前向传播算法类似，可以对随机上下文无关文法推导出一个更通用的版本，推导过程留做练习［这也被称为“内部”（inside）算法］。用类似于HMM中Viterbi算法的动态规划方法，可以得到根据SCFG的序列的最可能分析树。同样，我们仍将使用“Viterbi分析树”或“Viterbi推导”的这两个概念。值得注意的是，考虑到SCFG比HMM更复杂，需要采用三维形式的动态规划，因此它的复杂度是$O(N^3)$而不是$O(N^2)$（进一步的细节见参考文献［460,156］）。

11.6 学习算法

在参考文献［25,345,459,460,156］中，描述了一类SCFG的学习算法和其他算法。对于HMM，第一层次的贝叶斯推断的基本思想是通过用某种循环算法将似然度或后验概率最大化，从而估计模型参数。在上面提到的大多数例子中，这是通过某种形式的EM算法完成的，当然也可以使用其他方法，如梯度下降法。因为其推导与在HMM中详细描述过的方法平行，所以在此只概述每条学习规则。为简单起见，我们从ML估计开始，存在一条训练序列O，SCFG的规则和参数w已知。该方法可以直接扩展到MAP估计和多个训练序列。考虑一般的产生式规则$u \rightarrow \beta$。对O的任何推导π，定义$n(\beta, u, \pi, O)$为π中$u \rightarrow \beta$规则的使用次数。类似地，我们令$n(u, \pi, O)=\sum_{\beta} n(\beta, u, \pi, O)$

11.6.1 EM算法

对算法的E步骤，令$Q(\pi)=\mathbf{P}(\pi|O, w)$。如果用$P_{u \rightarrow \beta}$表示与规则对应的概率参数，则EM的参数重新估计的公式为

$$P^{+}_{u \rightarrow \beta}=\frac{\sum_{\pi} Q(\pi) n(\beta, u, \pi, O)}{\sum_{\pi} Q(\pi) n(u, \pi, O)}=\frac{\sum_{\pi} \mathbf{P}(\pi|O, w) n(\beta, u, \pi, O)}{\sum_{\pi} \mathbf{P}(\pi|O, w) n(u, \pi, O)}=\frac{n_{u \rightarrow \beta}}{n_{u}} \qquad (11.12)$$

这个重新估计公式很简单：所有的复杂性都隐藏在分子和分母的计算中。可以用动态规划过程计算它们，和11.5节中讨论过的类似，也和HMM的前向—后向算法类似，其计算量以$O(N^3)$增长，而HMM以$O(N^2)$增长，其中N表示序列的平均长度。

如果其中的文法是乔姆斯基范式，这个算法就是著名的内部—外部（inside-outside）算法。对有K个训练序列$O_1,\cdots,O_K$的情况，EM重估计公式为

$$P_{u\to\beta}^{+}=\frac{\sum_{j=1}^{K}\sum_{\pi}\mathbf{P}\left(\pi|O_j,w\right)n\left(\beta,u,\pi,O_j\right)}{\sum_{j=1}^{K}\sum_{\pi}\mathbf{P}\left(\pi|O_j,w\right)n\left(u,\pi,O_j\right)} \tag{11.13}$$

参考文献［460］给出了一种用于SCFG的EM算法，称为树文法EM算法（tree-grammar EM）。它的优点是计算量按$O(N^2)$增长，但需要折叠结构的RNA作为训练样本。折叠结构比原始序列提供了更多的信息，但比完整的分析树提供的信息少。如果有完整的分析树可用，只要统计每个产生式规则出现的次数就可以了。另一方面，折叠结构提供了树的骨架，叶节点标上了序列中的字符，但内部的节点没有标签。从骨架可以判断哪些核苷酸是配对的，但无法直接判断某个字符是由匹配生成的还是由插入非终结符生成的。树文法EM算法估计了非终结符对应的概率。

全局的循环训练算法也可能实现，如在参考文献［460］中，在第一步中，当前的文法被用于折叠训练序列，而在第二步中，折叠序列被用于优化文法的参数——如使用树文法EM算法。可以在文法中增加和删去产生式规则，就像在标准HMM体系中调整长度的算法那样。

表11-3 tRNA家族上使用SCFG RNA模型的测试结果

数据集	tRNA类型	总数	ZeroTrain	MT10CY10	MT100	Random TRNA618
ARCHAE	古细菌	103	0	0	0	50
CY	细胞质	230	0	10	0	100
CYANELCHLORO	蓝色体和叶绿体	184	0	0	0	100
EUBACT	真细菌	201	0	0	0	100
VIRUS	病毒	24	0	0	0	10
MT	线粒体	422	0	10	100	200
PART Ⅲ	第Ⅲ部分	58	0	0	0	58
总　数		1 222	0	20	100	618

11.6.2 梯度下降法和Viterbi学习算法

就目前所知，在SCFG的文献中只有EM算法已经得到了应用，但显然可以使用其他学习算法，例如梯度下降法和Viterbi学习算法。（对于复杂的SCFG来说，模拟退火算法的运算量还是太大。）

如同HMM，我们可以对SCFG进行参数重估：

$$P_{u\to\beta}=\frac{e^{w_{u\to\beta}}}{\sum_{\gamma}e^{w_{u\to\gamma}}} \tag{11.14}$$

于是梯度下降的在线学习公式是

$$\Delta w_{u\to\beta}=\eta(n_{u\to\beta}-n_u P_{u\to\beta}) \tag{11.15}$$

其中η是学习率。而在SCFG的Viterbi学习算法中，$n(\beta, u, \pi, O)$对所有可能的推导π取的平均值被与最可能推导π^*对应的$n(\beta, u, \pi^*, O)$代替。HMM中关于梯度下降法和Viterbi学习算法的大部分结论，经过适当修改后都适用于SCFG。从折叠序列出发的Viterbi学习算法本质上与用已有的多重序列比对初始化SCFG等价。

11.7 SCFG的应用

和在第7章与第8章中使用的HMM一样，训练过的SCFG可以用在很多同样的地方。对每个样本序列，可以计算它的Viterbi分析树。对RNA序列，它的语法结构或等价的分析树提供了可能的最优折叠，这可以用于预测其二级结构。这个方法是对前面基于系统进化分析或热力学的RNA二级结构预测方法的补充。分析树也可以用于推导多重序列比对，对齐的列或成对的列对应于非终结主状态。缺口也必须用这种显而易见的方法引入。这有助于确定共同的模式。可以对任何序列计算负对数似然度（或后验）分值。和HMM一样，序列的分值依赖于其长度，需要进行归一化，第8章中讨论过这个问题。这些分值可以用于区分家族的成员和非成员，可以用于搜索数据库，还可能发现家族的新成员。生成模式的SCFG可用于推测给定家族的新成员，虽然这点还没有经过检验。最后，SCFG可以模块化组合。参考文献［156］中讨论了一个例子：一个tRNA的SCFG与一个内含子文法组合起来，被用于搜索tRNA基因。

表11-4　使用四种文法所预测的二级结构中与原始碱基对匹配相符合的百分比

数据集	ZeroTrain	MT10CY10	MT100	Random TRNA618
ARCHAE	94.87	100.00	100.00	100.00
CY	98.28	99.76	99.89	99.87
CYANELCHLORO	96.22	99.64	99.64	99.79
EUBACT	99.69	99.86	99.86	99.86
VIRUS	96.83	100.00	100.00	100.00
MT	89.19	98.33	98.91	98.93
PARTⅢ	55.98	81.10	83.21	83.00

11.8 实 验

这里给出了参考文献［460］中tRNA家族的SCFG RNA模型的检验结果。参考文献［156］给出了类似的结果。原始数据集中包含1 222个独立的tRNA的序列和比对，这些序列和比对是从参考文献［502］所提供的数据库中提取的。它们的长度介于51到93个碱基之间，序列按照不同的tRNA类型分成7个不相交的集合（表11-3）。

为了进行识别实验，根据GenBank中序列的non-tRNA（包括mRNA，rRNA和蛋白质编码区）特征表产生了2 016个non-tRNA测试序列，对20到120之间的每个长度，大约生成了20个non-tRNA序列，然后生成了四个不同的文法。第一个文法（ZeroTrain）作为控制组，没有在任何序列上训练过，只包含tRNA的先验信息。另外三个文法（MT10CY10，MT100，Random TRNA618）是从表11-3所示的不同集合中训练出来的，使用的是树文法EM算法。需要在三个方面上比较这四个文法：多重序列比对、二级结构预测及识别。

表11-5 每个tRNA家族中，能与非tRNA区分开的tRNA数目及其对应的识别阈值

	5σ以上				4σ~5σ				4σ以下			
数据集	ZT	MT10	MT100	R618	ZT	MT10	MT100	R618	ZT	MT10	MT100	R618
ARCHAE	66	103	103	103	19	0	0	0	18	0	0	0
CY	135	230	230	230	53	0	0	0	42	0	0	0
CYANELCH	61	184	184	184	52	0	0	0	71	0	0	0
EUBACT	160	201	201	201	30	0	0	0	11	0	0	0
VIRUS	16	24	24	24	4	0	0	0	4	0	0	0
MT（训练集）	N/A	10	99	193	N/A	0	1	6	N/A	0	0	1
MT（测试集）	64	389	313	218	89	10	7	3	269	13	2	1
PART Ⅲ	0	9	7	29	1	15	14	8	57	34	37	21
NON-TRNA	0	0	0	0	0	0	1	1	2 016	2 016	2 015	2 015
总 数	502	1 150	1 161	1 182	248	25	23	18	2 488	2 063	2 054	2 038

11.8.1 多重序列比对

分别使用四个文法，对数据集中的全部1 222个tRNA序列进行比对。其中Random TRNA618得到了最好的结果。预测的比对与数据集原始的比对基本吻合（图11-3）。螺旋和环的边界是相同的。主要的区别在于额外臂，其长度和序列都是高度可变的。参考文献［460］中给出了文法比对对原始的比对做出小改进的例子。

```
              [      ]  <       D 区         > <   反义密码子    ><       ><   T 区        >[       ]
              (((((((  ((((               )))) (((((   ===  )))))       (((((        ))))))))))))
1 DC0380  -GCCAAGGTGGCAGAGTTCGGCCTAACGCGGCGGCCTGCAGAGCCGCTC----ATCGCCGGTTCAAATCCGGCCCTTGGCT---
2 DA6281  -GGGCGTGTGGCGTAGTC-GGT--AGCGCGCTCCCTTAGCATGGGAGAG----GTCTCCGGTTCGATTCCGGACTCGTCCA---
3 DE2180  --GCCCCATCGTCTAGA--GGCCTAGGACACCTCCCTTTCACGGAGGCG----A-CGGGGATTCGAATTCCCCTGGGGGTA---
4 DC2440  -GGCGGCATAGCCAAGC--GGT--AAGGCCGTGGATTGCAAATCCTCTA----TTCCCCAGTTCAAATCTGGGTGCCGCCT---
5 DK1141  -GTCTGATTAGCGCAACT-GGC--AGAGCAACTGACTCTTAATCAGTGG---GTTGTGGGTTCGATTCCCACATCAGGCACCA
6 DA0260  -GGGCGAATAGTGTCAGC-GGG--AGCACACCAGACTTGCAATCTGGTA----G-GGAGGGTTCAAGTCCCTCTTTGTCCACCA
7 DA3880  -GGGGCTATAGTTTAACT-GGT--AAAACGGCGATTTTGCATATCGTTA----T-TTCAGGATCGAGTCCTGATAACTCCA---
8 DH4640  -AGCTTTGTAGTTTATGTG-----AAAATGCTTGTTTGTGATATGAGTGAAAT-------------------TGGAGCTT---
              (((((((  ((((               )))) (((((    ===  )))))      (((((        ))))))))))))
1 DC0380  -GCCAAGGUGGCAG.AGUUcGGccUAACGCGGCGGCCUGCAGAGCCGCUC---AUCGCCGGUUCAAAUCCGGCCCUUGGCU---
2 DA6281  -GGGCGUGUGGCGU.AGUC.GG..UAGCGCGCUCCCUUAGCAUGGGAGAGG---UCUCCGGUUCGAUUCCGGACUCGUCCA---
3 DE2180  -GCCCC-AUCGUCU.AGAG.GCc.UAGGACACCUCCCUUUCACGGAGGCG----ACGGGGAUUCGAAUUCCCCU-GGGGCU--A
4 DC2440  -GGCGGCAUAGCCA.AGC-.GG..UAAGGCCGUGGAUUGCAAAUCCUCUA---UUCCCCAGUUCAAAUCUGGGUGCCGCCU---
5 DK1141  -GUCUGAUUAGCGC.AACU.GG..CAGAGCAACUGACUCUUAAUCAGUGGG---UUGUGGGUUCGAUUCCCACAUCAGGCACCA
6 DA0260  -GGGCGAAUAGUGUcAGCG.GG..-AGCACACCAGACUUGCAAUCUGGUA----GGGAGGGUUCGAGUCCCUCUUUGUCCACCA
7 DA3880  -GGGGCUAUAGUUU.AACU.GG..UAAAACGGCGAUUUUGCAUAUCGUUA----UUUCAGGAUCGAGUCCUGAUAACUCCA---
8 DH4640  -AGCUUUGUAGUUU.A--U.GU..GAAAAUGCUUGUUUGUGAUAUGAGUGA--AAU-----------------UGGAGCU
```

图11-3 数据集里的几条代表性的tRNA的多重序列比对（上）[502]
与训练过的随机TRNA618文法产生的结果（下）比较

"( )"指出了配对碱基位置；"==="为反义密码子；"[ ]"为接纳体螺旋的5'和3'端。对于Random TRNA618，大写字母对应于与文法的匹配非终结符对齐的碱基；小写字母对应子插入；"-"对应于通过跳跃生成的删除；"."填充插入对应的字符位置。序列取自上面的七个组，并标以数据库代码：1. ARCHAE（*Halobacterium cutirubrum*）；2. CY（*Saccharomyces cerevisiae*）；3. CYANELCHLORO（*Cyanophora paradoxa*）；4. CYANELCHLORO（*Chlamydomonas reinhardtii*）；5. EUBACT（*Mycoplasma capricolum*）；6. VIRUS（*Phage T5*），7. MT（*Aspergillus nidulans*）；8. PARTⅢ（*Ascaris suum*）。

11.8.2 RNA二级结构预测

对于二级结构预测，在大多数情况下，Viterbi分析树能给出正确的二级结构。表11-4给出了原始比对中的碱基对在每个文法所预测的二级结构中出现的百分比。对ARCHAE和VIRUS，三个经过训练的文法都达到了100%的识别；对CY、CYANELCHLORO和EUBACT也有非常好的一致性；对PART Ⅲ则要差很多。

11.8.3 识别

测试四个文法的识别能力的方法，是计算所有2 016个non-tRNA序列的标准化分值，并与数据集里的1 222个tRNA序列的分值相比较。non-tRNA序列的标准分值很少超过4，所以识别阈值被定为5。表11-5给出了每个家族中能成功地与non-tRNA区分开的tRNA数目。图11-4显示了一些对应的柱状图。

即使只用很少的20个序列训练，也可以显著地提高检测率，比较MT10CY10和ZeroTrain可以看出这一点。除了对MT和PARTⅢ这两个子集外，MT10CY10完全正确地把tRNA和non-tRNA区分开了，而在前两个子集上，ZeroTrain文法失败了。MT10CY10在MT子集上识别得相当好，但在PARTⅢ上不行。MT10CY10识

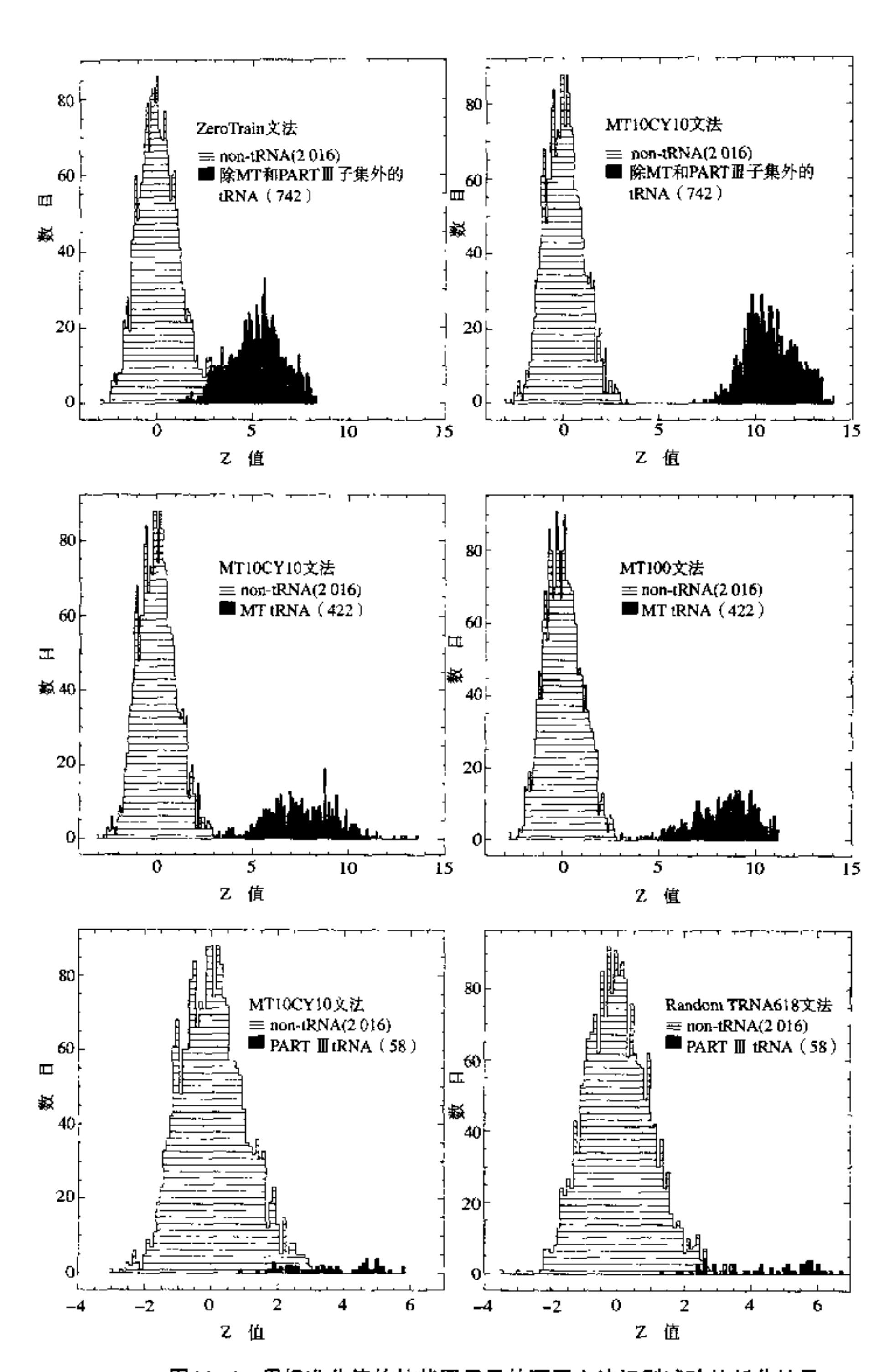


图11-4　用标准分值的柱状图显示的不同文法识别试验的部分结果

别了422条线粒体序列中的399条，这个识别水平几乎和其他用更多tRNA序列训练出来的文法差不多。对于PARTⅢ序列，没有一个文法能识别得比较好，即使是RandomTRNA618也不行，虽然它在这些序列的一部分上训练过。在PARTⅢ序列上的训练能提高文法在这些序列上的识别水平，但仍有一半序列得到的标准分值低于阈值5。

11.9 展　望

至此我们已经回顾了形式语言和文法的基本理论，知道了如何通过推广骰子模型和HMM的思想把随机文法应用到生物序列上。特别是，SCFG和对应的学习算法已经被应用于推导tRNA的统计模型。训练过的文法被用于比对、折叠和识别tRNA,并且取得了很好的结果。SCFG方法是确定tRNA二级结构的可行方法。它补充了原有的两种方法，一种基于对同源序列的系统进化分析，[186,565, 278]另一种则基于热力学。[521,222,527,585]然而，与应用于蛋白质家族研究的HMM相比，用于RNA的SCFG还没有被足够彻底地检验过，还需要更多的工作来进一步确立这种方法。SCFG能找出RNA的全局结构比对，而最近出现了一个求局部结构比对的新动态规划算法，该算法效果良好。[220,221]这个局部方法是对Smith-Waterman比对方法的扩展，并和另一种动态规划技术结合，寻找最大数量的互补碱基对。

本章的文法方法有一些局限。首先，它们计算量很大，因此对于长的序列尤其是$N \geq 200$的序列，现在的算法仍有一些不切实际。第二，并非所有的RNA结构可以由SCFG得到。它的分析树不能反映伪结和非成对的相互作用等其他三级相互作用，所有这些都被忽略了。第三，本章所描述的方法不包括内含子的模型，而某些tRNA基因中存在内含子。这些限制为未来的研究指出了一些显而易见的方向，包括：

- 算法或者硬件的速度提升；
- 建立文法，可能是图文法（或其他模型）以及对应的训练算法，以便引入RNA三级结构或者其他分子的三级结构；
- SCFG的模块化结合（如同HMM），用来对更复杂的RNA序列建模，包括对内含子建模，这方面的工作可见参考文献［156］；
- 对更大和更具挑战性的RNA序列建模，例如rRNA序列；
- 最后，沿着第9章的思路发展SCFG／NN（或SG／NN）的混合构架，其中NN用于计算SCFG的参数，或者用于调整、混合几个不同的SCFG。

第12章　微阵列和基因表达

12.1　微阵列数据简介

在过去短短几年里，基于微阵列的新技术大量涌现并且不断飞速发展。这一类技术包括DNA杂交阵列（hybridization array）［基因表达阵列以及用于测序和多态性研究的寡核苷酸（oligonucleotide）阵列］、蛋白质阵列、组织阵列以及组合化学阵列等。由于这些高通量方法使大量分子与一个大型文库之间的组合反应成为可能，这些方法很快就产生了数以兆计的信息，远远超过传统的生物分析方法所提供的数据。这一章主要讨论研究DNA基因表达的微阵列技术。紧随参考文献［44］的思路，我们将介绍如何系统地将一般性的概率体系用于分析微阵列数据，更完整的有关DNA微阵列处理的介绍参见文献［43］。

DNA基因表达微阵列使得生物学家能够在基因组层次上研究任何种类细胞在任何时间、任何给定条件下的基因表达模式。[148,160,263]在这类微阵列中，所有的RNA被反转录成带有放射性同位素或荧光标记的cDNA。然后，cDNA与由基因片断组成的、固定在玻片或膜上的大型DNA文库杂交。最后，采用荧光或其他成像技术测定上千个基因在各种不同实验条件下的表达。利用这些微阵列，人们正产生出大量的数据，它们可以帮助我们深入地认识诸多生物过程的本质，如基因功能、发育、癌症、衰老和药理等。[498,567,7,217,354,511,7,554,369,169,171]即使是对现有信息的部分理解也能够提供很有价值的线索。例如，新基因的共表达（co-expression）就可以帮助我们推断许多缺乏相关信息的基因的功能。然而，基因微阵列数据分析方法的发展现在才刚刚起步。[581]

基因表达微阵列数据至少可以在三个复杂性依次递增的层次上进行分析。第一个层次是单基因层次，主要研究单个基因在处理条件和对照条件下是否有不同的表达。第二个层次是多基因层次，主要从共同功能、相互作用、共调控等角度研究基因族。在第三个层次上，人们则试图推测出隐藏在我们观察到的基因表达模式背后的基因或蛋白质调控网络。

首先，为简化起见，假设每个基因X对应了两组测量值$X_1^c,\cdots,X_{n_c}^c$和$X_1^t,\cdots,X_{n_t}^t$，分别是该基因在对照条件和处理条件下的表达水平或其对数。在这里，处理条件是广义的，它指的是任何不同于对照的条件。对每个基因来说，所关注的基本问题是在这两种条件下，基因的表达水平有无显著不同。虽然从表面上看，用标准的统计方法可以轻松地解决这个问题，但是实际情况并非如此。

一种常用的方法是一种简单的倍数法。在这种方法中，只要对照组和处理组的基因平均表达水平的比值超过了一个常数因子——通常为2，就认为基因的表达水平发生了显著变化。然而，对基因表达数据的仔细分析表明，用这个简单的"2倍法则"不太可能得到最优的结果，因为在表达谱的不同区域，"2"这个因子所代表的显著性是大不相同的。

解决这个问题的另一种方法是使用t检验法，例如对表达水平的对数值进行t检验。这有点类似于倍数法，因为对数的差和比值的对数是相等的。但是由于均值的对数不等于对数的均值，这两种方法并不等价。事实上，由于对数函数的凸性，均值的对数总是严格大于对数的均值。但是通过合理近似，检验两个基因对数表达水平差异的显著性，与检验它们比值的变化与1是否有显著差异是等价的。

在t检验中，利用经验均值m_c、m_t和方差s_c^2、s_t^2，按照下式计算两个总体之间的标准化距离：

$$t=(m_c-m_t)/\sqrt{\frac{s_c^2}{n_c}+\frac{s_t^2}{n_t}} \tag{12.1}$$

其中，$m=\sum_i x_i/n$和$s^2=\sum_i (x_i-m)^2/(n-1)$是我们所熟知的对总体均值和总体方差的估计。从统计学文献中不难知道，t近似服从自由度为

$$f=\frac{\left[\left(s_c^2/n_c\right)+\left(s_t^2/n_t\right)\right]^2}{\frac{\left(s_c^2/n_c\right)^2}{n_c-1}+\frac{\left(s_t^2/n_t\right)^2}{n_t-1}} \tag{12.2}$$

的学生氏分布（附录A）。如果t超过了某个由给定的置信水平确定的阈值，两个总体就被认为是不同的。在t检验中，由于总体均值之间的距离被经验标准差标准

化，因此可以避免固定阈值倍数法的一些缺点。用t检验法分析微阵列数据的根本问题在于实验重复的次数n_c和（或）n_t往往太少，这是因为即便采用最新的技术，反复做同一个实验仍然是相当昂贵或者乏味的。n=1,2或3的小样本的情况依然非常常见，这会导致一些问题，例如方差会被明显地低估。因此，需要一个更好的理论体系来解决所有这些不足。

12.2　阵列数据的概率模型

12.2.1　高斯模型

由于阵列数据噪声大、波动大，而且在大量数据的背后还有很多相关变量不能被观测到，因此需要用一种概率方法来处理这种数据。为了理解处理阵列数据的概率方法，应该先回忆一下前面所学过的序列数据的分析方法。在第3章中讲到，序列数据的最简单概率模型是一个反映DNA、RNA或蛋白质序列家族的平均组成的概率骰子。下一个复杂层次上的模型是一个1阶马尔可夫模型。在这个模型中，序列的每个位置或者多重比对的每一列都对应了一个概率骰子。在前面我们已经看到，这些模型虽然简单，但作为背景模型，它们是非常有用的，比如我们可以用它们来评估更复杂的模型的性能。

在阵列数据处理中，最简单的模型假设所有的数据点相互独立，并且从同一个连续分布如高斯分布中抽取。这样一个"高斯骰子"模型仍然需要计算一些我们感兴趣的量，例如平均活性水平及其标准差。这些量在刻画或估计数据的全局特性时有用。与序列数据的建模相当，更复杂的模型是一组独立分布，每个分布对应一维变量，比如一个基因。由于基因之间存在着复杂的相互作用关系，它们显然不是独立的。尽管如此，独立性这一近似假设还是有用的。事实上，任何用概率或其他方法逐个基因地确定表达水平差异显著性的尝试或可能性，都建立在这一假设的基础之上。

这里首先假设在某种给定条件下，一个基因的表达水平的测量值大致服从高斯分布。根据经验，就通常采用的技术而言，这一假设是合理的，特别是表达水平的对数大致服从对数正态分布。就目前所知，还没有进行大规模的重复实验来获得更精确的估计。如果真的进行重复实验，诸如伽玛分布、高斯/伽玛混合分布的其他分布显然也会被引入。引入这些分布将会影响分析的细节（见参考文献［558,403］），然而一般性的贝叶斯概率体系不变。

因此，下面假定数据都经过了预处理——包括在必要时取了对数——使得一

个基因在一种条件（处理或对照）下的表达水平的测量值可以用一个正态分布$\mathcal{N}(x;\mu,\sigma^2)$来建模。对每个基因和每个条件，都对应有一个双参数模型$w=(\mu,\sigma^2)$。通过把目光集中在这样一个模型上，可以忽略基因和实验条件本身的标记。假设观测结果是独立的，则似然函数由下式给出：

$$\begin{aligned}\mathbf{P}(D|\mu,\sigma^2)&\approx\prod_i\mathcal{N}(x_i;\mu,\sigma^2)\\&=C(\sigma^2)^{-n/2}\exp[-\sum_i(x_i-\mu)^2/2\sigma^2]\\&=C(\sigma^2)^{-n/2}\exp\{-[n(m-\mu)^2+(n-1)s^2]/2\sigma^2\}\end{aligned}\tag{12.3}$$

其中，i取遍所有的重复测量。在本章中，采用C来表示任何分布的归一化常数（$C=1/Z$）。似然度仅仅取决于充分统计量n、m和s^2。换句话说，样本的所有与似然度相关的信息都包含在这三个数里了。高斯模型的均值或方差已知的情形相对比较简单，参考文献［86,431］对此有全面的讨论。

完整的贝叶斯处理还需引入一个先验分布$\mathbf{P}(\mu,\sigma^2)$。如何选取这个先验分布是建模的一部分。有几种可能的选择，[86,431]这反映了贝叶斯方法的灵活性，但它绝不是随意的。采用共轭先验分布既方便又能充分体现DNA微阵列数据的一些性质，其中包括我们将要看到的一个性质，即μ和σ^2不是独立的。

12.2.2 共轭先验分布

当先验分布和后验分布的函数形式相同时，该先验分布称为共轭先验分布。对于方差已知的正态分布，当我们估计其均值时，共轭先验分布显然也是一个正态分布。我们已经看到，在生物序列的骰子模型中，标准的共轭先验分布是一个Dirichlet分布。在（12.3）中，似然函数的形式表明共轭先验密度一定也具有$P(\mu|\sigma^2)P(\sigma^2)$的形式，其中边缘分布$P(\sigma^2)$对应一个标定逆伽玛分布（相当于$1/\sigma^2$服从伽玛分布，见附录A），而条件分布$P(\mu|\sigma^2)$是正态分布。

由此可以导出一个分层模型，在该模型中，先验分布的四个超参数构成一个向量$\alpha=(\mu_0,\lambda_0,\nu_0,\sigma_0^2)$，先验分布的密度为

$$\mathbf{P}(\mu|\sigma^2)=\mathcal{N}(\mu;\mu_0,\sigma^2/\lambda_0)\tag{12.4}$$

和

$$\mathbf{P}(\sigma^2)=\mathcal{I}(\sigma^2;\nu_0,\sigma_0^2)\tag{12.5}$$

当且仅当$v_0>2$时，先验分布的期望是有限的。先验分布$\mathbf{P}(\mu, \sigma^2)=\mathbf{P}(\mu, \sigma^2|\alpha)$由下式给出：

$$C\sigma^{-1}\left(\sigma^2\right)^{-(v_0/2+1)}\exp\left[-\frac{v_0}{2\sigma^2}\sigma_0^2-\frac{\lambda_0}{2\sigma^2}\left(\mu_0-\mu\right)^2\right] \quad (12.6)$$

注意，对于微阵列数据，采用一个μ和σ^2相互不独立的先验分布很有意义，只要查看典型的基因微阵列数据集，立刻就能看出这一点（图12-1）。超参数μ_0和σ^2/λ_0可以看做μ的位置和尺度，超参数v_0和σ_0^2可以看做σ^2的自由度和尺度。经过一些代数运算，可以推导出后验分布具有与先验分布相同的函数形式：

$$\mathbf{P}(\mu, \sigma^2|D, \alpha)=\mathcal{N}(\mu; \mu_n, \sigma^2/\lambda_n)\mathcal{I}(\sigma^2; v_n, \sigma_n^2) \quad (12.7)$$

其中

$$\mu_n=\frac{\lambda_0}{\lambda_0+n}\mu_0+\frac{n}{\lambda_0+n}m \quad (12.8)$$

$$\lambda_n=\lambda_0+n \quad (12.9)$$

$$v_n=v_0+n \quad (12.10)$$

$$v_n\sigma_n^2=v_0\sigma_0^2+(n-1)s^2+\frac{\lambda_0 n}{\lambda_0+n}\left(m-\mu_0\right)^2 \quad (12.11)$$

后验分布的参数以一种合理的方式将先验分布的信息和数据信息结合了起来。均值μ_n是先验均值和样本均值的凸加权平均。后验自由度v_n是先验自由度加上样本量。后验平方和$v_n\sigma_n^2$是先验平方和$v_0\sigma_0^2$、样本平方和$(n-1)s^2$及残差不确定度（residual uncertainty）的总和。残差不确定度由先验均值和样本均值之间的差异确定。

虽然可以对基因表达数据采用一个先验均值μ_0，但在很多情况下，令$\mu_0=m$就足够了。随之可以准确地得到后验平方和，就好像有v_0个额外的观测，而每个额外的观测都具有偏差σ_0^2一样。虽然从表面上看，这好像是在观测到数据以后再设置先验分布，[372]但是通过预先设定一个μ_0，并使$\lambda_0\to 0$，可以得到类似的效果。此时，由于标准差很大，均值位置的先验分布几乎是一个均匀分布。先验分布的超参数选择将在下面详细讨论。

不难验证，均值的条件后验分布$P(\mu|\sigma^2, D, \alpha)$是正态分布$\mathcal{N}(\mu_n, \sigma^2/\lambda_n)$。均值的边缘后验分布$P(\mu|D, \alpha)$是学生氏分布$t(v_n, \mu_n, \sigma_n^2/\lambda_n)$，方差的边缘后验分布$P(\sigma^2|D, \alpha)$是标定逆伽玛分布$\mathcal{I}(v_n, \sigma_n^2)$。

如果μ和σ^2相互独立，即$\mathbf{P}(\mu, \sigma^2)=\mathbf{P}(\mu)\mathbf{P}(\sigma^2)$，且它们的先验分布在函数形式上与共轭先验分布相同（分别是正态分布和标定逆伽玛分布），则也可以采用半共轭先验分布。但是正如前面所讨论的，独立性假设对于DNA微阵列数据并不是太适用。正如利用共轭先验分布的混合分布可以得到共轭后验分布的混合分布，也可以利用混合分布构造出更复杂的先验分布。

12.2.3 参数点估计

后验分布$\mathbf{P}(\mu, \sigma^2|D, \alpha)$是贝叶斯分析的基本对象，它包含了$\mu$和$\sigma^2$所有可能取值的相关信息。但是，为了进行前文所描述的t检验，需要把这个信息含量丰富的分布函数浓缩成基因在某种给定条件下的表达水平的均值和方差的单点估计。我们可以通过多种途径进行估计。一般来说，用后验估计的均值（MP）得到的答案是最鲁棒的。另一种选择是用后验分布的众数，也就是MAP（最大后验）估计。为了讨论的全面性，以下分别来推导这两种估计。

若$v_n>2$，MP估计可以通过积分由下式给出：

$$\mu=\mu_n \quad \text{和} \quad \sigma^2=\frac{v_n}{v_n-2}\sigma_n^2 \tag{12.12}$$

若取$\mu_0=m$且$v_0+n>2$，则可以得到下面的MP估计：

$$\mu=m \quad \text{和} \quad \sigma^2=\frac{v_n\sigma_n^2}{v_n-2}=\frac{v_0\sigma_0^2+(n-1)s^2}{v_0+n-2} \tag{12.13}$$

这就是下面将要提到的Cyber-T软件采用的缺省估计(default estimate)。由(12.7)，MAP估计是

$$\mu=\mu_n \quad \text{和} \quad \sigma^2=\frac{v_n\sigma_n^2}{v_n-1} \tag{12.14}$$

如果我们取$\mu_0=m$，它们就可以化简成

$$\mu=m \quad \text{和} \quad \sigma^2=\frac{v_n\sigma_n^2}{v_n-1}=\frac{v_0\sigma_0^2+(n-1)s^2}{v_0+n-1} \tag{12.15}$$

这里，边缘后验分布的众数由下式给出：

$$\mu=\mu_n \quad \text{和} \quad \sigma^2=\frac{v_n\sigma_n^2}{v_n+2} \tag{12.16}$$

实际上，（12.13）和（12.15）给出的结果相近，它们都可以用于基因表达阵列。这两种方法的细微差别在于，一般来说（12.13）稍好一些，这点我们在第3章讨论序列数据的Dirichlet先验分布时就已经看到了。Dirichlet先验分布相当于引入伪计数（pseudo-count），从而避免将某种氨基酸或核苷酸的概率设为0。在阵列数据中，通过较少的观测点给出的方差估计结果可能会很差。例如对一个单点（n=1），我们当然想避免将方差设为0，因此需要通过共轭先验分布做正则化处理。在MP估计中，利用v_0个具有背景方差σ_0^2的伪观测（pseudo-observation）对经验方差进行调整。

12.2.4　完整的贝叶斯处理和超参数点估计

到了建模的这个阶段，每个基因都与两个模型$w_c=(\mu_c, \sigma_c^2)$和$w_t=(\mu_t, \sigma_t^2)$，两组超参数α_c和α_t，两个后验分布$\mathbf{P}(w_c|D, \alpha_c)$和$\mathbf{P}(w_t|D, \alpha_t)$相关联。完整的概率处理需要引入超参数的先验分布。可以对超参数的先验分布积分得到真实后验分布$\mathbf{P}(w_c|D)$和$\mathbf{P}(w_t|D)$，然后再对这两个后验分布沿w_c和w_t积分以判断两个模型是否不同。注意这一方法明显要比单纯的t检验法更具一般性，原则上它可以发现t检验法检测范围之外的有趣变化。例如，如果一个基因的表达水平在对照条件和处理条件下均值相等但方差差别很大，哪怕方差的这种差别有可能与具体的生物现象有关，t检验法也检测不到这种差别。即使我们只考虑μ_c和μ_t且它们服从高斯后验分布，我们也必须从数值上估计出$\mathbf{P}(|\mu_c-\mu_t|<\varepsilon)$。虽然利用现在的计算机求解后者并不是什么难事，但是我们也完全可以采用仅仅依赖于点估计的方法，这种方法比完整的贝叶斯处理更简单，近似程度更好。

然而，点估计需要超参数，这可以通过很多途径解决。[372,375]一种途径仍然是定义超参数的一个先验分布，然后试着对它们积分计算出真实后验分布$\mathbf{P}(w|D)$并确定后验分布的众数，从而得到w的真实MAP估计。更确切地说，这需要求$\mathbf{P}(w|\alpha)$和$\mathbf{P}(w|\alpha D)$对超参数向量α的积分。一个可以取代对超参数积分的方法是在参考文献［372］中描述的显著性理论框架（evidence framework）。在显著性理论框架中，我们用后验分布的MAP估计计算超参数的点估计（MP还是需要对超参数积分）：

$$\mathbf{P}(\alpha|D)=\frac{\mathbf{P}(D|\alpha)\mathbf{P}(\alpha)}{\mathbf{P}(D)} \qquad (12.17)$$

如果我们取先验分布$\mathbf{P}(\alpha)$为均匀分布，这就等价于最大化显著性$\mathbf{P}(D|\alpha)$：

$$\begin{aligned}\mathbf{P}(D|\alpha)&=\mathbf{P}(D|w, \alpha)\mathbf{P}(w|\alpha)/\mathbf{P}(w|D, \alpha)\\&=\mathbf{P}(D|w)\mathbf{P}(w|\alpha)/\mathbf{P}(w|D, \alpha)\end{aligned} \qquad (12.18)$$

原则上，计算显著性需要对模型参数w积分。但是，利用似然函数、共轭先验分布和后验分布来表示，我们可以避免积分而从（12.18）求出：

$$\mathbf{P}(D|\alpha)=(2\pi)^{-n/2}\frac{\sqrt{\lambda_0}}{\sqrt{\lambda_n}}\frac{(v_0/2)^{v_0/2}}{(v_n/2)^{v_n/2}}\frac{\sigma_0^{v_0}}{\sigma_n^{v_n}}\frac{\Gamma(v_n/2)}{\Gamma(v_0/2)} \tag{12.19}$$

参考文献［44］讨论了显著性的偏导数和临界点，证明了众数$\mu_0=m$。

12.2.5 贝叶斯假设检验

本质上，迄今为止对于每个基因在每种条件下的对数表达水平，都是用高斯分布来建立的模型。如果所关心的仅仅是一个给定的基因有没有发生改变，我们可以直接对处理组和对照组的基因对数表达水平的差建模。这些差值可以两两考虑或成对考虑，这更接近于目前微阵列技术的做法：沿着两个不同的通道（红和绿），测量在处理条件和对照条件下表达水平的比值的对数。

依旧可以用高斯分布$\mathcal{N}(\mu, \sigma^2)$对$x^t-x^c$建模。在给定的数据下，设定零假设$H$为$\mu=0$（无变化）。为了避免将零概率赋予零假设，这里的贝叶斯方法必须赋予$\mu=0$一个非零的先验概率，这可能显得有些随意。无论如何，根据前面对共轭先验分布的推导，可以设$\mathbf{P}(\sigma^2)=\mathcal{I}(\sigma^2; v_0, \sigma_0^2)$。对均值$\mu$，我们用混合密度

$$\mu=\begin{cases} 0 & : \quad p \\ \mathcal{N}(0, \sigma^2/\lambda) & : \quad 1-p \end{cases} \tag{12.20}$$

参数p可以通过前面的试验固定，或者作为一个超参数处理，比如可以使之服从Dirichlet分布。相关统计量$\log[\mathbf{P}(\bar{H})/\mathbf{P}(H)]$的计算作为练习留给读者。

12.2.6 实　现

为了提高效率，使用名为Cyber-T⑫的网络服务器系统实现了一个折中的解决方法。[44,366] 在这一方法中，采用了t检验法，其中标准差按照（12.13）进行了正则化，自由度则与相应增加的样本总体相关，该自由度偶尔可能取分数维。在Cyber-T中，无论是对原始数据还是对对数化以后的数据都可以采用单纯的或引入贝叶斯方法的t检验。

在最简单的情况下，取$\mu_0=m$，用户必须选择背景方差σ_0^2及其强度v_0。参数v_0表示的是背景方差对经验方差σ_0^2的置信度。v_0的值可以由用户设定。n越小，v_0应

⑫ 可访问http://128.200.5.223/CyberT/。

该越大。一个简单的法则是：为了合理地估计标准差，需要$l>2$个点，并保证$n+v_0=l$。在不同基因的数据点数目n不同时，用这种方法处理数据有很大的灵活性。一个合理的缺省值（default）是$l=10$。如果基因的活性水平接近于所使用技术的最低检测水平，那么可以把它作为一种特殊情况处理。此时，基因的测量值非常不可信，用较大的v_0值赋予它们较强的先验概率也许更明智一些。

对σ_0可以采用所有观测的标准差，也可以根据情况采用某类基因的标准差。有一种灵活的实现方法，即用所有包含在大小为ws的窗里的相邻基因，估计背景标准差。Cyber-T对所有基因的表达水平进行自动排序，用户可以自行选择窗的大小。ws的缺省值为101，对应于所研究基因上下各50个基因。也可以自适应地调节窗的大小和用回归估计求σ_0^2。

12.2.7 仿 真

我们已经用贝叶斯方法和Cyber-T分析了大量已发表的和未发表的数据集。在每个我们分析过的高密度微阵列实验中都观察到：在重复实验中，表达水平的方差明显随均值伸缩（无论是在原始尺度上还是在做了对数变换以后）。其结果是，仅仅根据变化倍数来确定显著性阈值，对于表达水平低的基因来说过于随意，而对于表达水平高的基因来说又过于保守。虽然使用旧的方法也取得了一些具有生物意义的相关结果，但我们发现贝叶斯方法还是比简单的倍数法或直接的t检验法要好一些，它用一致的统计方法部分地克服了由于实验重复次数少造成的缺陷。[366]

如果要比较贝叶斯方法和简单的t检验法或倍数法，参考文献［19］给出的高密度阵列实验是一个能够提供很多信息的数据集，这个实验比较了大肠杆菌野生型和突变型细胞的球形调控蛋白IHF（整合宿主因子）。这个数据集的主要优点是它对野生型和突变型的等位基因都进行了四次重复实验。基于背景标准差的先验分布的正则化效果可以参见图12-1和下面描述的仿真。此图清楚地显示标准差在表达水平的取值范围内有很大变化，大致随表达水平的增加单调递减，当然人们也观察到其他情况。有趣的是，在这些图里，表达水平低的基因对应的对数表达水平方差要比表达水平高的基因对应的方差大。这些图证实了低水平或接近背景水平表达的基因也许需要一个比较大的v_0，或者干脆在表达分析中忽略它们。低水平表达基因测量值的方差是如此之大，以至于在很多情况下难以检测出以这种水平表达的基因有任何显著变化。

在分析数据时，我们常常发现这样的情况，即表达水平的变化倍数很大，但在贝叶斯分析中，相应的p值并不意味着在统计上有任何变化。相反地，表达水平的变化倍数很小，但在贝叶斯分析中，这种变化往往被认为具有很高的显

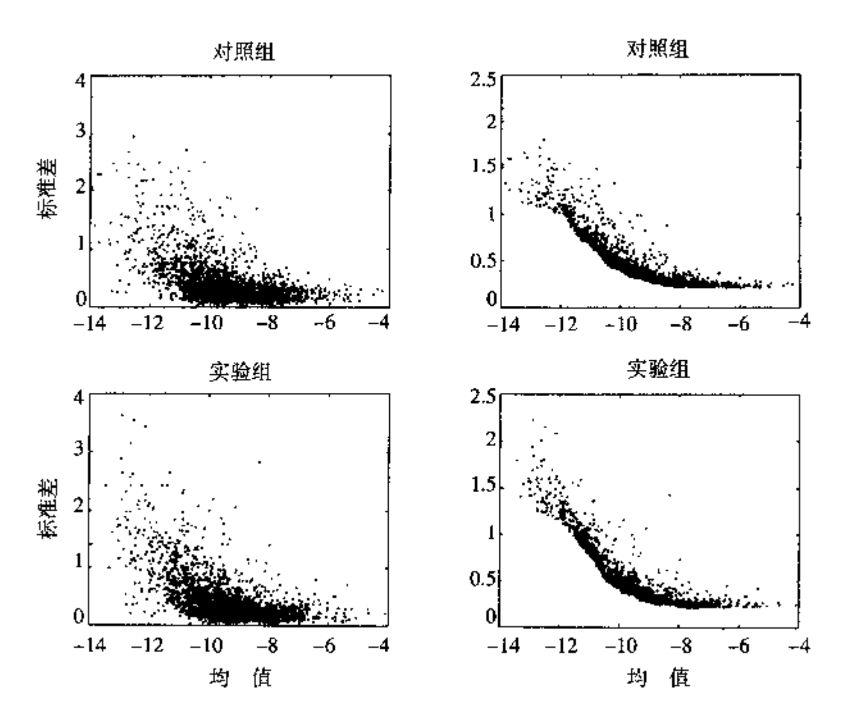


图12–1 大肠杆菌的DNA微阵列实验

数据由P^{33}标记的逆转录RNA与包含大肠杆菌所有4 290个预测基因的商业化尼龙微阵列（Sigma Genosys）杂交获得。样本包括一个野生株（对照）和一个球形调控蛋白——整合宿主因子（IHF）基因缺失株（处理）。在对照和实验两种情况下均有n=4。水平轴表示表达水平的对数的均值μ，垂直轴表示相应的标准差（σ）。左边一列对应的是原始数据，右边一列对应的是按照（12.13）进行正则化的标准差。窗口大小ws=101，l=10（见正文）。数据来自参考文献［19］。

著性。在这两种情况下，贝叶斯方法的结论比单纯倍数法的结论显得鲁棒性更强一些，这是由于统计上不显著的大倍数变化常常与大的测量误差联系在一起，而统计上显著但变化倍数小于2倍的基因常常能够被非常准确地测量。在参考文献［19］这样的实验重复水平上，可以比较贝叶斯估计方法与t检验的一致性。我们发现从IHF数据集中抽取的样本数目为2的独立样本集中（即2个实验样本对2个对照样本），用贝叶斯方法得到的120个最显著基因集合有大约50%的基因相同，而用t检验方法得到的120个最显著基因集合仅有大约25%的基因相同。这表明做2次重复实验，用贝叶斯方法确定基因是否被上调或下调，其结果的一致性大约是简单t检验法的2倍。尽管对高密度微阵列实验来说，只做2次实验将会存在很大的不确定性。

为了进一步评估贝叶斯方法，可以假设对数表达水平服从高斯分布，其均值

和方差的取值范围与参考文献［19］中数据集的均值和方差范围相似，然后人为地产生一个数据集，其中对每一种参数组合产生1 000组数据。选定对数数据的均值和标准差（括号内）的组合如下：−6(0.1)，−8(0.2)，−10(0.4)，−11(0.7)，−12(1.0)。利用这些仿真数据，我们可以比较简单倍数法（2倍和5倍）、简单t检验法和采用了Cyber-T缺省设置的贝叶斯t检验法的性能。表12−1给出了主要结果，归纳如下：

表12−1　从1 000个基因中检测出的阳性数目

	对数表达水平		比　值		简单的t检验法		贝叶斯方法	
n	从	到	2倍	5倍	$p<0.05$	$p<0.01$	$p<0.05$	$p<0.01$
2	−8	−8	1	0	38	7	73	9
2	−10	−10	13	0	39	11	60	11
2	−12	−12	509	108	65	10	74	16
2	−6	−6.1	0	0	91	20	185	45
2	−8	−8.5	167	0	276	71	730	419
2	−10	−11	680	129	202	47	441	195
3	−8	−8	0	0	42	9	39	4
3	−10	−10	36	0	51	11	39	6
3	−12	−12	406	88	44	5	45	4
3	−6	−6.1	0	0	172	36	224	60
3	−8	−8.5	127	0	640	248	831	587
3	−10	−11	674	62	296	139	550	261
5	−8	−8	0	0	53	13	39	8
5	−10	−10	9	0	35	6	31	3
5	−12	−12	354	36	65	11	54	4
5	−6	−6.1	0	0	300	102	321	109
5	−8	−8.5	70	0	936	708	966	866
5	−10	−11	695	24	688	357	752	441
2v4	−8	−8	0	0	35	4	39	6
2v4	−10	−10	38	0	36	9	40	3
2v4	−12	−12	446	85	46	17	43	5
2v4	−6	−6.1	0	0	126	32	213	56
2v4	−8	−8.5	123	0	475	184	788	509
2v4	−10	−11	635	53	233	60	339	74

数据根据对数尺度上的正态分布，在Arfin等人2000年发表的数据的范围内产生，对每组参数产生1 000组数据。对数数据的均值及其对应的标准差（括号内）如下：−6(0.1)，−8(0.2)，−10(0.4)，−11(0.7)，−12(1.0)。对n的每个取值，头三个实验对应无变化的情况，由此产生假阳性率。分析由Cyber-T在缺省参数(ws=101，l=10)下完成，自由度为$n+v_0-2$。

- 若有5次重复实验（5个对照和5个处理），贝叶斯方法和t检验法的结果相似。
- 如果重复实验的次数很少（2或3次），贝叶斯方法比t检验法效果好。
- 贝叶斯方法和t检验法的假阳性率与期望一致（分别为0.05和0.01）。而当实验的重复次数很少（也就是2次）时，贝叶斯方法的假阳性率会提高。
- 比值的假阳性率是表达水平的函数，在低表达水平时要高得多。在低表达水平时，比值的假阳性率高得难以接受。
- 如果给定重复实验的次数，$p<0.01$的贝叶斯方法能比2倍变化法检测出更多的表达差异，但在低表达水平时除外（此时比值的假阳性率被提高了）。
- 重复实验2次的贝叶斯方法的性能优于重复实验3次的t检验法（或者是2次对4次）。
- 贝叶斯方法在比较3个处理对3个对照，或2个处理对4个对照时性能相似。这提示了一种实验方法，即通过多次重复对照实验减少处理实验的重复次数。

12.2.8 更复杂的概率模型

我们已经发展了阵列数据分析的概率体系，解决了现有方法的很多不足，这些不足与小样本偏差以及如下事实有关，即在不同的表达水平下，不同倍数的差异具有不同的重要性。这一概率体系是用高斯分布的基因独立模型进行分层贝叶斯建模的一种形式。虽然高斯分布是否合理需要进一步验证，但我们可以很容易地将其他分布引入到一个类似的体系中。虽然没有什么方法比重复实验好（见参考文献［355］），但仿真实验和受控的重复实验[366]已经表明，这种方法对数据有一种正则化效果，它比传统的t检验法或简单的倍数法要好，可以部分地补偿重复实验的缺乏。

根据具体目标和算法实现，这一方法可以在很多方面加以推广。例如，可以离线计算回归函数用以建立标准差和表达水平之间的关系，并用以产生背景标准差。这一方法还可以自适应地调整窗口的大小来计算局部背景方差，例如窗口的大小可以由回归函数的导数决定。在标准差变化相对平缓的区域内（也就是图12-1中-8~-4的范围），窗口较小；在标准差快速变化的区域内（也就是图12-1的-12~-10的范围），窗口较大。也可以实现一种更完整的贝叶斯方法，例如对边缘后验分布（在这里是学生氏分布）积分来估计概率$\mathbf{P}(\mu_c \approx \mu_t | D, \alpha_t, \alpha_c)$。

这一方法还可以扩展成更复杂的方案，以及（或者）包含了实验变量梯度的方案和（或）针对时间序列的方案。其中一个例子是针对以下情况设计的方案：

细胞在不同种类的刺激下（尿素、氨、过氧化物），或者在同一种刺激但刺激的摩尔浓度不同（0、5、10mM）的条件下生长。一般化的线性和非线性模型可以用于这种环境。然而，最富挑战性的问题在于如何将这里的概率体系扩展到第二层次的分析，把基因之间可能的作用和相关性都考虑进来。如果两个或多个基因在某种处理条件下表现出相似的行为，那么在它们对应的基因族层次上所做的关于表达变化的处理方法将会具有更强的鲁棒性。多维正态模型和高斯过程（附录E）提供了这一层次分析的概率模型。

举个例子，对于多维正态模型，μ是均值向量，正定对称矩阵Σ定义了协方差矩阵，其行列式为$|\Sigma|$。似然函数有如下形式：

$$C|\Sigma|^{-n/2}\exp\left[-\frac{1}{2}\sum_{i=1}^{n}(X_i-\mu)^t\,\Sigma^{-1}(X_i-\mu)\right]. \qquad (12.21)$$

将归一化标定逆伽玛分布加以推广，得到基于逆Wishart分布（附录A）的共轭先验分布。逆Wishart分布实际上是对标定逆伽玛分布的推广，它给Σ提供了一个先验分布。与一维的情形类似，这个共轭先验分布具有参数（$\mu_0,\Lambda_0/\lambda_0,\nu_0,\Lambda_0$）。$\Sigma$服从参数为$\nu_0$和$\Lambda_0^{-1}$ Wishart分布。在给定Σ的条件下，μ服从多维正态先验分布$\mathcal{N}(\mu;\ \mu_0,\ \Sigma/\lambda_0)$。后验分布具有与先验分布相同的形式，是多维正态分布和逆Wishart分布的积，参数为$(\mu_n,\Lambda_n/\lambda_n,\nu_n,\Lambda_n)$。参数满足

$$\begin{aligned}
\mu_n &= \frac{\lambda_0}{\lambda_0+n}\mu_0+\frac{n}{\lambda_0+n}m\\
\lambda_n &= \lambda_0+n\\
\nu_n &= \nu_0+n\\
\Lambda_n &= \Lambda_0+\sum_{1}^{n}(X_i-m)(X_i-m)^t\\
&\quad+\frac{\lambda_0 n}{\lambda_0+n}(m-\mu_0)(m-\mu_0)^t.
\end{aligned} \qquad (12.22)$$

对于多维情形，也可以推导出类似于（12.13）的估计。

虽然多维正态和其他相关模型可以提供一个较好的起点，但是对影响阵列数据的高阶效果建立好的概率模型仍然处于发展的起步阶段。迄今为止，大多数方法都或多或少地集中在聚类方法的专门应用上。

12.3 聚类

12.3.1 概述

在复杂性的下一层次上，我们想去掉一些简化的假设，例如所有基因都是独立的这个假设。我们将从这里开始看基因的协方差矩阵，看那里是否存在相关基因族等。聚类分析不仅可以用于处理阵列数据，而且可以用于处理生物信息学的其他很多问题，包括一些序列分析的问题。因此这里将试图简要而全面地介绍聚类，讨论有时不局限于阵列数据的分析。

聚类是探索性数据分析和模式发现的一种基本手段，其目的是提取数据中隐含的类别结构。但是，聚类是一个模糊的概念，它没有一个准确的定义。已知有几十种聚类算法和大量的专门的聚类程序被用于DNA微阵列数据的分析，其类型涵盖了分级聚类、k均值聚类等，[160,7,253,511,484,124,194]它们没有一个显而易见的共同点。由于聚类问题的多样性和“开放性”，不大可能给出聚类的一个系统化的完备处理框架。在聚类问题和聚类算法中，特别是在基因表达条件下，有很多重要问题需要考虑。

数据类型

在最高层次上，聚类算法可以根据被聚类数据的性质加以区分。在标准情况下，数据点是欧几里德空间内的向量，但这绝不是惟一的可能。除了向量数据或用绝对坐标表示的数值型数据以外，在某些情况下，数据可用相对坐标表示，此时给定的是任意两点之间的距离。在很多情况下，数据用两两相似程度（或不相似程度）表示，这种度量常常不能满足距离的三个公理性假设（非负性、对称性和三角不等式）。还存在这样的情况，即数据轮廓用三重的或更高阶次的关系表示，或只给出了所有可能的两两相似度的一个子集。更重要的是，在有些情况下，数据既不是向量型的也不是关系型的，而是定性的，例如调查问卷多项选择题的答案。有时也把这称做名词性数据。虽然目前基因表达阵列数据主要是数值型的，但这种情况在将来有可能会发生变化。实际上，“与基因正交”的维涵盖了不同的实验、不同的病人、不同的组织和不同的时间等，它至少在某些部分不是数值型的。随着阵列数据数据库的增长，在很多情况下，数据混杂了向量型数据和名词性数据。

有监督聚类/无监督聚类

聚类算法之间的一个重要区别在于它们是有监督的还是无监督的。在有监督

聚类中，聚类基于一个给定的参考向量集或类别集。在无监督聚类中，没有一个事先定义的向量集或类别集。混合方法也是可能的，可以先进行无监督聚类，紧跟着再进行有监督聚类。目前，基因表达阵列实验尚处于早期阶段，像k均值和自组织映射[511]这样的非监督聚类方法是最常用的。但也有人尝试过有监督的聚类，他们用功能信息或非监督聚类方法预先确定一些族，然后用神经网络或支持向量机（附录E）这类可以学习数据类别之间决策边界的分类器，将新基因归到不同的族内。[194]

相似度

一些聚类算法（包括几种形式的分级聚类算法）的起点是聚类对象的一个两两相似度矩阵。准确地定义相似度非常关键，可以在很大程度上影响聚类算法的输出。举个例子，在序列分析中，相似度可以用一个对间隙和替换打分的分值矩阵以及一个比对算法来定义；在基因表达的分析中可以应用不同的相似度定义。两个明显的例子是欧几里德距离（或者更一般的L^p距离）和表达水平向量的相关性。Pearson相关系数就是两个标准化向量的点积，或它们夹角的余弦。它可以用于度量不同实验条件或不同时间点下的基因对。根据情况的不同，每种相似性度量都有各自的优缺点，因而或多或少适合于某种分析。例如，相关性能够反映形状的相似性但不强调两组测量的数值关系（magnitude），而对偏差（outlier）十分敏感。再比如考虑测量两个在背景表达水平附近涨落的不相关基因，这样的基因若以欧几里德距离来度量非常相似（距离接近于0），但若以相关性来度量却很不相似（相关性接近于0）。类似地，考虑两个向量1000000000和0000000001。在某种意义上它们很相似，因为它们都几乎为0。但在另一方面，由于分别在头尾两个位置存在偏差，它们的相关性接近于0。

类别数

聚类的类别数K的选择是一个非常棘手的问题，它取决于我们在什么尺度上观察数据。虽然有人尝试过发展自动确定类别数的方法，[484]但是可以肯定，通过半手工的方法反复试验仍然是最有效的一种确定类别数的手段，现阶段对微阵列数据来说尤其如此。

代价函数和概率表述

任何对给定数据集聚类的严格讨论，都要预先给出一种原则性的方法，来比较同一数据集的不同聚类结果，这样就需要某种形式的易于计算的全局代价/误差函数。这样，聚类的目标就变成了最小化这一函数。这也被称为参数化聚类，与

之相对的是非参数聚类，后者只有一些局部函数可以采用。[72]

一般来说，至少对数值型数据而言，这一函数依赖于以下一些量：类中心、类内各点与相应类中心的距离、类内平均相似度等。对于数据的聚类结果，这样一个函数往往是不连续的。这里依然没有普遍适用的函数，代价函数（cost function）必须根据具体的问题来确定，不同的代价函数会导致不同的结果。

鉴于概率方法和概率建模的优点，将聚类的代价函数与相应的概率模型的负对数似然度相结合，这个想法很吸引人。虽然这从形式上总是可能的，但是我们最关心的还是：在什么时候隐含概率模型的结构及其相关的独立性假设是清晰的。也就是说什么时候代价函数中的求和项反映了隐含的概率和变量的因子结构。我们将要看到，混合模型正是如此。在混合模型中，k均值聚类算法可以看做EM算法的一种形式。

在余下的部分里，我们将更详细地讨论两种基本的聚类算法：分级聚类和k均值聚类，它们都可以用于DNA微阵列数据分析。其他很多相关方法，包括矢量量化、[104,484]主成分分析、因子分析、自组织映射、神经网络和SVM等可以在参考文献中找到。

12.3.2 分级聚类

聚类可以通过分级的分支过程得到。因此有一些方法，可以根据两两相似度从数据中自动建立一棵树。对于基因表达的情况，这就是参考文献［160］所用的方法。这种方法的输出是一棵树而非一组类别。特别地，如何从树中定义类别往往不明显，因为类别是通过在树的某些点剪枝得到的，而这一过程或多或少带有随意性。

参考文献［160］用的标准算法从相关（pr距离或相似度）矩阵C开始迭代计算一个标准树图，把所有元素集成到一棵树里。在算法的每一步：

- 计算当前矩阵中两个最相似的元素（具有最大的相关性），生成一个节点将它们结合在一起。
- 通过求两个元素表达谱（expression profile）（或向量）的平均（缺失的数据可以忽略，求平均时可以按照向量中元素的个数进行加权）生成新节点的表达谱（或向量）。也可以不计算表达谱，而用距离的加权平均来估计新的类中心之间的距离。
- 用新节点取代两个结合的元素，按照新计算的表达谱（或向量）计算新的相关矩阵。这个相关矩阵比原来的矩阵要小一些。
- 从N个点开始，这个过程将最多重复N-1次，直至只剩下1个单节点。

生物学家很熟悉这个算法，它被用于序列分析、系统进化树和平均连锁聚类分析。正如已经指出的，在建立了这样一个标准树图以后，如何显示结果以及如何选取类别仍然是个问题。在每个节点上，两个由节点结合在一起的元素都可以被排到另一个元素的右边或左边。由于有$N-1$次结合，与树的结构一致的线性排列的总数为2^{N-1}。一个最优的线性排列要使排列中所有相邻的一对节点的结合相似度的总和达到最大，但一般来说我们不能有效地计算出这样一种最优排列。参考文献［160］用了一种启发式近似算法，它用平均表达水平、染色体位置和最大诱导时间（time of maximal induction）对基因加权。通过对一组基因表达数据聚类得到的主要类别确实显示出了生物学上的相关性。

12.3.3 k均值聚类法、混合模型和EM算法

k均值聚类法

在所有的聚类算法中，k均值聚类法[153]作为针对隐含混合模型的EM算法的一种形式，可能具有最清晰的概率表述。在k均值聚类法的一种典型实现中，类别数被固定为一个值K。一开始就给各类选择代表点或类中心，这样K个代表点或类中心的选择或多或少带有随意性。它们也被称做质心（centroid）或原型（prototype）。然后在每一步：

- 把每个点分到离它最近的代表点所代表的类内；
- 分类后计算新的代表点，比如取每一个新类的平均或重心；
- 重复上面两个步骤，直到系统收敛或涨落很小。

因此我们要注意：k均值聚类法要求选择类别数，要求可以计算点与点之间的距离或相似度，并且对于每一类在给定其成员时可以计算代表点。

当代价函数与一个隐含的概率混合模型[172,522]对应时，k均值聚类法是经典EM算法的一种在线近似，而且它一般会收敛到一个解，这个解至少是一个局部ML或MAP解。一个经典情况是在混合高斯模型中应用欧几里德距离。参考文献［28］给出了它在序列聚类中的一个相关应用。

混合模型和EM算法

为了更进一步理解这一点，设想一个数据集$D=(d_1,\cdots,d_N)$和一个隐含的混合概率模型，该模型有K个分量，分量具有如下形式：

$$\mathbf{P}(d)=\sum_{k=1}^{K}\mathbf{P}(M_k)\mathbf{P}(d|M_k)=\sum_{k=1}^{K}\lambda_k\mathbf{P}(d|M_k) \qquad (12.23)$$

其中$\lambda_k\geqslant 0$，$\sum_k\lambda_k=1$，且M_k是第k类的模型。与对数似然度相联系的拉格朗日函

数和对混合系数的归一化约束由式

$$\mathcal{L}=\sum_{i=1}^{N}\log\left(\sum_{k=1}^{K}\lambda_k\mathbf{P}\left(d_i|M_k\right)\right)-\mu\left(\sum_{k=1}^{K}\lambda_k-1\right) \tag{12.24}$$

和相应的临界方程

$$\frac{\partial\mathcal{L}}{\partial\lambda_k}=\sum_{i=1}^{N}\frac{\mathbf{P}\left(d_i|M_k\right)}{\mathbf{P}\left(d_i\right)}-\mu=0 \tag{12.25}$$

给出。将每个临界方程乘以λ_k，然后对k求和，马上就可以得到拉格朗日算子$\mu=N$。再将每个临界方程乘以$\mathbf{P}(M_k)=\lambda_k$，用如下形式的贝叶斯定理

$$\mathbf{P}(M_k|d_i)=\mathbf{P}(d_i|M_k)\mathbf{P}(M_k)/\mathbf{P}(d_i) \tag{12.26}$$

就可以得到

$$\lambda_k^{*}=\frac{1}{N}\sum_{i=1}^{N}\mathbf{P}(M_k|d_i) \tag{12.27}$$

因此，第k类的混合系数的ML估计是d_i来自模型k的条件概率的样本均值。现在考虑每个模型M_k有它自己的参数向量(w_{kj})的情况。将拉格朗日函数对w_{kj}求导得到

$$\frac{\partial\mathcal{L}}{\partial w_{kj}}=\sum_{i=1}^{N}\frac{\lambda_k}{\mathbf{P}\left(d_i\right)}\frac{\partial\mathbf{P}\left(d_i|M_k\right)}{\partial w_{kj}} \tag{12.28}$$

将（12.26）代入（12.28），对每个k和j，最终可得临界方程

$$\sum_{i=1}^{N}\mathbf{P}(M_k|d_i)\frac{\partial\log\mathbf{P}\left(d_i|M_k\right)}{\partial w_{ki}}=0 \tag{12.29}$$

用于估计参数的ML方程，是分别从每个点得到的ML方程$\partial\log\mathbf{P}(d_i|M_k)/\partial w_{kj}=0$的加权平均。和（12.27）一样，权重值是$d_i$隶属于每一类的概率。

和HMM一样，可以迭代使用ML方程（12.27）和（12.29）来搜索ML估计，这也给出了EM算法的另一个例子。在E步骤，对每个数据点，估计它对每个混合分量的隶属概率（隐变量）。M步骤相当于K个不同的估计问题，每个数据点对与每个分量相关联的对数似然度都有一定贡献，这个贡献被估计出的隶属概率加权。根据隶属概率$\mathbf{P}(M|d)$在E这一步是以何种方式估计（硬估计还是软估计）的，同

样的算法可以有不同的结果。上面给出的k均值聚类法使用硬估计，其隶属概率要么是0，要么是1，每个点只能属于一个类别。这类似于用HMM的EM算法的Viterbi版本，其中只用到了与一个序列相关的最优路径，而不是所有可能路径构成的家族。算法的M这一步也有几种不同的版本，例如参数w_{kj}可以使用梯度下降法估计，也可以通过精确求解（12.29）得到。众所周知，一个点集的重心能够最小化该集合与任何固定点的平方距离。因此在球形高斯分量的混合模型中，前面描述的k均值聚类法的M这一步，最大化了相应的对数似然度的平方，给出了每个高斯分量均值的ML估计。

也可以对每一族的参数及（或）混合系数引入如下形式的先验分布

$$\mathbf{P}(d)=\sum_{k=1}^{K}\mathbf{P}(d|M_k, w_k)\mathbf{P}(w_k|M_k)\mathbf{P}(M_k) \tag{12.30}$$

这将导致更复杂的分级概率模型，它们在处理DNA阵列数据甚至是序列数据时，可能会更有用。例如在序列数据中，这就相当于用不同的骰子产生序列，骰子来自于不同的工厂，工厂分散在不同的国家，等等；在每一层次上，对每种相应的属性都有一个概率分布。就我们所知，对于这类分级混合模型，还没有研究者系统地在这种环境中研究过。

12.4 基因调控

最后，就分析的第三个层次而言，DNA微阵列表达数据很自然地导致了基因调控的很多问题。在系统的层次上理解基因调控是生物学中最有趣同时也是最富挑战性的问题。有关这个问题的绝大部分原理还没有被发现。这里只提一些主要的研究方向并给出一些参考文献。

其中一个研究方向是对调控区进行数据挖掘，例如寻找转录因子的DNA结合位点和其他调控motif（一段具有特色功能的生物序列）。在一定程度上，这种搜索可以在基因组层次上用纯计算工具进行。[530,531,232]其基本思想是计算每种长度为N的词（N-mers）在全基因组或基因组的一个特定子集（如所有基因的上游区域）里出现的次数，N的值通常为3到10。出现次数超过一般水平的词称为超频词（overrepresented N-mers）。我们对超频词特别感兴趣，它们构成了很多已知的调控motif。超频词的分布也带有很多信息。[232]当然，在任何情况下，超频现象必须根据一个好的统计背景模型来评估，这样的背景模型可以是一个从实际计数中得来的具有一定阶次的马尔可夫模型。如果在此基础上还有基因表达数据可用，

数据挖掘过程还可以做进一步调整，例如在给定条件下被上调（up-regulated）［或下调(down-regulated)］的基因的上游区内寻找超频现象。[89,231,535,111,270] 由于motif结构和位置的多变性，像EM和吉布斯采样这样的概率算法，很自然会在motif的寻找中扮演很重要的角色（参考MEME和CONSENSUS之类的程序）。不管怎么说，目前按照这种方法找到的motif，只有一小部分能在TRANSFAC[560] 数据库和现有文献中找到，其他大部分有待于未来实验检验。

另一个更深人的研究方向是试图在全局层次上，或从某些特定的局部[532,190,584]（如一条代谢途径或一组共调控基因）对调控网络进行建模或推断。这里的一个主要障碍是我们尚不清楚转录在分子水平上的所有细节。例如，我们还不完全理解噪声在基因调控中扮演的角色。[383,243] 而且目前具备详细信息的调控网络还很少，况且这样的调控网络看上去都非常复杂。[579] 在理论方面，一些数学范式已经被用于基因网络建模。这不仅包括离散模型，如布尔网［考夫曼（Kauffman）的先驱性工作[310,311,312]］，还包括了基于微分方程的连续模型，如连续反馈神经网络[391] 或power-law范式[537,466,258]、概率图模型和贝叶斯网络。[190] 但没有一个范式能够反映基因调控的所有变量，这一领域的大部分工作还等着人们去做。有关这一活跃领域的更多文献，可以在过去几年的ISMB、PSB和RECOMB会议文集中找到。在系统层次上理解生物（例如参考文献［88,309,239,289,576］中的研究），基因网络、蛋白质网络、信号网络、代谢网络以及免疫系统或神经网络这样的特定系统可能都是今后几十年生物信息学要努力解决的中心问题。

第13章　互联网资源与公共数据库

13.1　迅速积累的资源

众所周知，在信息处理领域，因特网上的资源更新比几乎其他所有信息过程的变化都要快。生物序列分析的专用工具也是如此。新的工具不断产生并投入使用，现有的工具逐渐过时。而在生物信息学中的许多专业领域中，计算分析作为一个强大工具逐渐替代了实验研究的许多重要部分。

因特网上所提供的许多工具并不是由一些大机构或研究组织开发的，而是由一些个人研究者开发的，他们当中的许多人都只在短期内活跃于这一研究领域。资金的情况每年也有所不同，甚至对一些主要的计算服务机构也是如此。这意味着一些链接不能得到经常更新，许多服务器也不能够每天24小时运转。如果一项服务很受欢迎，那么它的服务器通常会得到充分及时地更新。但是在许多情况下，一些联合机构所建立的镜像服务替代了主服务器的更新，如华盛顿特区的NCBI、英国Hinxton的EBI和日本的DDJB。

因特网这个“开放式生物信息中心”的一个极其令人困惑的方面在于许多网站提供同一类型的服务，而这些服务基于不同的实现方法。例如，蛋白质的二级结构预测、基因发现和内含子的剪接位点预测就是这种情况。大多数网络还提供蛋白质中的氨基酸水溶性的预测。由于这些方法大都根据不同的数据集构造并测试，因此即使专家也很难客观地判断各种方法的优劣。通常，只使用一种特定的方法有许多不利之处，所以要遵循统计学的“平均结果优于单一选择”这一原则，综合多种方法才可以提高结果的鲁棒性和可信性。

建立算法评判基准是极其困难的，因为作为评判标准的序列集经常与构造某些算法所使用的序列集有很大程度的重叠。有些算法具有可以“记住”训练数据的内在机理，而另一些则仅设计为提炼数据集平均的或一般性的特征。对于这些算法，应用于训练集所表现出的性能必然优于它们在测试集中所表现出的性能。

正如第1章（第1.2节）中所描述的，序列数据总量呈指数增长。幸运的是，计算机或工作站的计算能力也呈指数增大，而且它们的价格一直在下降。很久以来，当计算机的价格降为大约原来的一半时，它的速度就会提升到原来的2倍。这意味着每6到10个月，从经济学角度来看，应用查询序列或正则表达式在公共数据库中进行同样的搜索所需的花费就会变为原来的2倍。这还意味着算法必须经常更新以维持搜索水平。

13.2 关于数据库和工具的综合目录

长期以来，在生物序列分析领域有一个传统做法，那就是建立已有的数据库

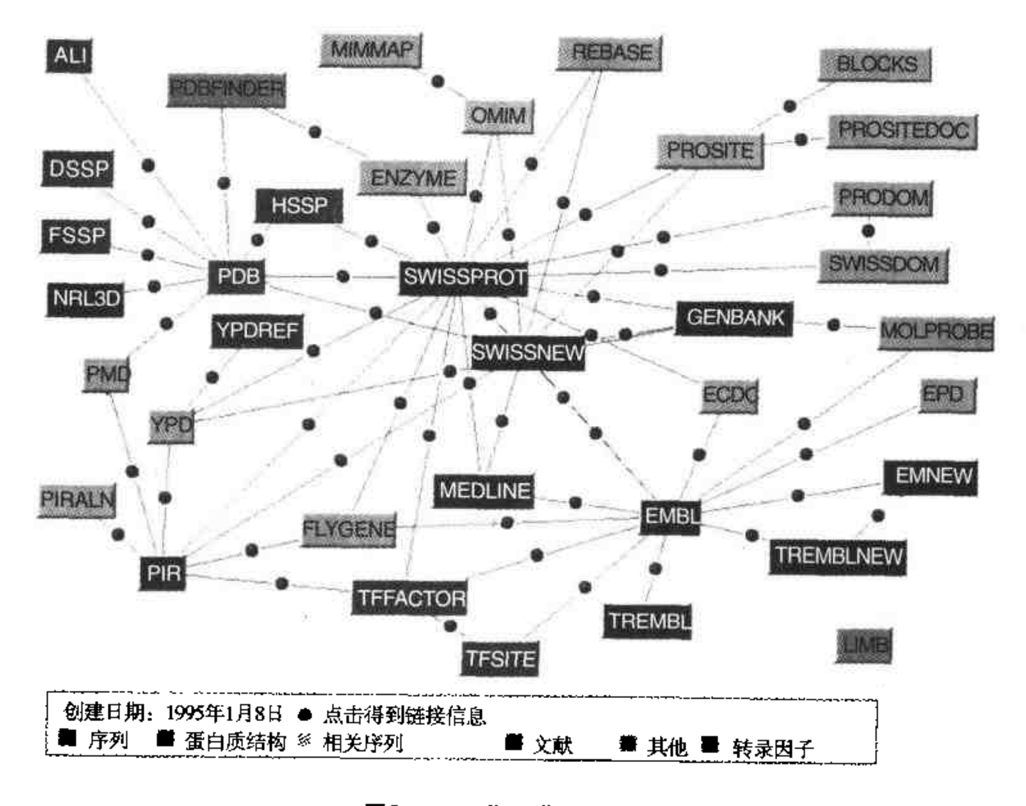


图3-1 一些互联网上可用的数据库

的综合目录（database over database），用于管理大量资源。最早期的这种数据库之一是LiMB（Listing of Molecular Biology database，分子生物学数据库列表），过去这一数据库通过硬拷贝发布。[353] 今天，更加灵活的万维网（WWW）方式成为传播数据的惟一合理的媒介。人们可以及时跟踪和更新网络链接。LiMB包括了与分子生物学有关的数据库维护信息。它的建立是为了使研究机构更方便地确定和找到他们所需的数据集。

下面各节包括了数据库综合目录、主要的公共序列数据库和有代表的预测服务器的链接列表。实际上，这些列表都应该是每日更新，我们的目的也不是提供WWW的详细指导。这些资料只是有助于对实验数据进行严格分析的诸多工具中的一些。我们推荐读者定期浏览一些原始的数据库，并使用一般的WWW搜索引擎寻找最新的资源。下面的大部分链接都来源于由丹麦生物序列分析中心的简·汉森（Jan Hansen）创建的网页（http://www.cbs.dtu.dk/biolink.html），它们主要关注序列和注释的检索，用于提交序列的网站没有包括在内。

13.3 分子生物学数据库综合目录

SRS序列查询系统（分子生物学数据库网络浏览器）

http://www.embl-heidelberg.de/srs5/

分子生物学数据库及服务器概览

http://www.ai.sri.com/people/pkarp/mimbd/rsmith.html

BioMedNet图书馆

http://biomednet.com

DBGET数据库链接

http://www.genome.ad.jp/dbget/dbget.links.html

哈佛基因组研究数据库与精选服务器

http://golgi.harvard.edu

约翰斯·霍普金斯大学（Johns Hopkins University）OWL网络服务器

http://www.gdb.org/Dan/proteins/owl.html

生物网络服务器索引，USCS

http://info.er.usgs.gov/network/science/biology/index.html

分子生物学数据库列表（LiMB）

gopher://gopher.nih.gov/11/molbio/other

病毒学的WWW服务器，UW-Madison

http://www.bocklabs.wisc.edu/Welcome.html

UK MRC人类基因组图谱计划研究中心

http://www.hgmp.mrc.ac.uk/

生物学家和生物化学家的WWW资源

http://www.yk.rim.or.jp/~aisoai/index.html

其他生物网络服务器的链接

http://www.gdb.org/biolinks.html

分子模型服务器与数据库

http://www.rsc.org/lap/rsccom/dab/ind006links.htm

EMBO实际结构数据库

http://xray.bmc.uu.se/embo/structdb/links.html

蛋白质科学家的网络资源

http://www.faseb.org/protein/ProSciDocs/WWWResources.html

ExPASy分子生物学服务器

http://expasy.hcuge.ch/cgi-bin/listdoc

抗体研究网页

http://www.antibodyresource.com

生物信息学网址

http://biochem.kaist.ac.kr/bioinformatics.html

乔治·梅森大学（George Mason University）的生物信息学与计算分子生物学专业

http://www.science.gmu.edu/~michaels/Bioinformatics/

INFOBIOGEN数据库目录

http://www.infobiogen.fr/services/dbcat/

国家生物技术信息研究室

http://www.nbif.org/data/data.html

人类基因组计划情报

http://www.ornl.gov/TechResources/Human_Genome

生物学软件及数据库档案

http://www.gdb.org/Dan/software/biol-links.html

蛋白质组研究：功能基因组学的新前沿（著作目录）

http://expasy.hcuge.ch/ch2d/LivreTOC.html

13.4 序列与结构数据库

13.4.1 主要的公共序列数据库

EMBL WWW服务器

http://www.EMBL-heidelberg.de/Services/index.html

GenBank数据库查询形式（得到GenBank的一个记录）

http://ncbi.nlm.nih.gov/genbank/query_form.html

蛋白质结构数据库WWW服务器（得到一个PDB结构）

http://www.rcsb.org

欧洲生物信息学研究中心（EBI）

http://www.ebi.ac.uk/

EBI产业支持

http://industry.ebi.ac.uk/

SWISS-PROT（蛋白质序列库）

http://www.expasy.ch/sprot/sprot-top.html

PROSITE（蛋白质功能位点）

http://expasy.hcuge.ch/sprot/prosite.html

大分子结构数据库

http://BioMedNet.com/cgi-bin/membersl/shwtoc.pl?J:mms

Molecules R Us（搜索及观察一个蛋白质分子）

http://cmm.info.nih.gov/modeling/net_services.html

PIR国际蛋白质序列数据库

http://www.gdb.org/Dan/proteins/pir.html

SCOP（蛋白质的结构分类），MRC

http: //scop.mrc-lmb.cam.ac.uk/scop/data/scop.1.html

洛斯阿拉莫斯（Los Alamos）的HIV序列数据库

http://hiv-web.lanl.gov/

洛斯阿拉莫斯的HIV分子免疫数据库

http://hiv-web.lanl.gov/immuno/index.html

TIGR数据库

http://www.tigr.org/tdb/tdb.html

NCBI WWW Entrez浏览器

http://www.ncbi.nlm.nih.gov/Entrez/index.html

剑桥结构数据库（小分子有机的及有机金属的结晶结构）

http://www.ccdc.cam.ac.uk

基因本体论坛

http://genome-www.stanford.edu/GO/

13.4.2 专业数据库

ANU生物信息学超媒体服务（病毒数据库、分类及病毒的命名法）

http://life.anu.edu.au/

O-GLYCBASE（O联糖基化蛋白质的修订数据库）

http://www.cbs.dtu.dk/ OGLYCBASE/cbsoglycbase.html

基因组序列数据序（GSDB）（已注释的DNA序列的关系数据序）

http://www.ncgr.org

EBI蛋白质拓扑图

http://www3.ebi.ac.uk/tops/ServerIntermed.html

酶及新陈代谢途径数据库（EMP）

http://www.empproject.com/

MAGPIE（多用途的基因组计划自动研究环境）

http://www.mcs.anl.gov/home/gaasterl/magpie.html

大肠杆菌数据库收集（ECDC）（大肠杆菌K12的DNA序列汇编）

http://susi.bio.uni-giessen.de/ecdc.html

嗜血流感杆菌数据库（HIDC）（遗传图谱，序列片断搜索目录）

http://susi.bio.uni-giessen.de/hidc.html

EcoCyc（大肠杆菌基因及其新陈代谢的百科全书）

http://www.ai.sri.com/ecocyc/ecocyc.html

Eddy实验室的snoRNA数据库

http://rna.wustl.edu/snoRNAdb/

GenProtEc（大肠杆菌基因及蛋白质）

http://www.mbl.edu/html/ecoli.html

NRSub（枯草芽孢杆菌的非冗余数据库）

http://pbil.univ-lyon1.fr/nrsub/nrsub.html

YPD（酿酒酵母蛋白质）

http://www.proteome.com/YPDhome.html

酵母基因组数据库

http://genome-www.stanford.edu/Saccharomyces/

LISTA、LISTA-HOP及LISTA-HON（酵母同源数据库汇编）

http://www.ch.embnet.org/

FLyBase（果蝇数据库）

http://flybase.bio.indiana.edu/

MPDB（分子探针数据库）

http://www.biotech.ist.unige.it/interlab/mpdb.html

tRNA序列及tRNA基因序列汇编

http://www.uni-bayreuth.de/departments/biochemie/trna/index.html

贝勒医学院（Baylor College of Medicine）的小RNA数据库

http://mbcr.bcm.tmc.edu/smallRNA/smallrna.html

SRPDB（信号识别粒子数据库）

http://psyche.uthct.edu/dbs/SRPDB/SRPDB.html

RDP（核糖体数据库计划）

http://rdpwww.life.uiuc.edu/

小核糖体亚蛋白RNA结构

http://rrna.uia.ac.be/ssu/index.html

大核糖体亚蛋白RNA结构

http://rrna.uia.ac.be/lsu/index.html

RNA修饰数据库

http://medlib.med.utah.edu/RNAmods/

HAMSTeRS（A型血友病突变数据库）及凝血因子VIII突变数据库

http://europium.csc.mrc.ac.uk/usr/WWW/WebPages/main.dir/main.htm

B型血友病（突变位点、短插入及删除序列）

ftp://ftp.ebi.ac.uk/pub/ databases/haemb/

人类p53基因，hprt及lacZ基因及其突变

http://sunsite.unc.edu/dnam/mainpage.html

PAH突变分析（导致人类苯丙酮尿症的苯丙氨酸强化酶特异位点）

http://www.mcgill.ca/pahdb

ESTHER（胆碱脂酶基因服务器）

http://www.ensam.inra.fr/cgi-bin/ace/index

IMGT（免疫遗传学数据库）

http://www.ebi.ac.uk/imgt/

人类肿瘤及细胞系的p53基因突变

ftp: //ftp.ebi.ac.uk/pub/databases/p53/

雄性激素受体基因突变数据库

ftp://www.ebi.ac.uk/pub/databases/androgen/

肾上腺皮质激素受体资源

http://nrr.georgetown.edu/GRR/GRR.html

甲状腺激素受体资源

http://xanadu.mgh.harvard.edu//receptor/trrfront.html

16SMDB及23SMDB（16S和23S核糖体RNA突变数据库）

http://www.fandm.edu/Departments/Biology/Databases/RNA.html

MITOMAP（人类线粒体基因组数据库）

http://www.gen.emory.edu/mitomap.html

SWISS-2DPAGE（二维凝胶电泳数据库）

http://expasy.hcuge.ch/ch2d/ch2d-top.html

PRINTS［蛋白质印迹（protein fingerprint）数据库］

http://www.biochem.ucl.ac.uk/bsm/dbbrowser/PRINTS/PRINTS.html

KabatMan（抗体结构及序列信息数据库）

http://www.bioinf.org.uk/abs/

ALIGN（蛋白质序列比对一览）

http://www.biochem.ucl.ac.uk/bsm/dbbrowser/ALIGN/ALIGN.html

CATH（蛋白质结构分类系统）

http://www.biochem.ucl.ac.uk/bsm/cath/

ProDom（蛋白质域数据库）

http://protein.toulouse.inra.fr/

Blocks数据库（蛋白质分类系统）

http://blocks.fhcrc.org/

HSSP（按同源性导出的蛋白质二级结构数据库）

http://www.sander.embl-heidelberg.de/hssp/

FSSP（基于结构比对的蛋白质折叠分类）

http://www2.ebi.ac.uk/dali/fssp/fssp.html

SBASE蛋白质域（已注释的蛋白质序列片断）

http://www.icgeb.trieste.it/~sbasesrv/

TransTerm（翻译控制信号数据库）

http://uther.otago.ac.nz/Transterm.html

GRBase（参与基因调控的蛋白质的相关信息数据库）

http://www.access.digex.net/~regulate/trevgrb.html

ENZYME（酶的命名法）

http://www.expasy.ch/enzyme/

REBASE（限制性内切酶和甲基化酶数据库）

http://www.neb.com/rebase/

RNaseP数据库

http://jwbrown.mbio.ncsu.edu/RNaseP/home.html

REGULONDB（大肠杆菌转录调控数据库）

http://www.cifn.unam.mx/Computational_Biology/regulondb/

TRANSFAC（转录因子及其DNA结合位点数据库）

http://transfac.gbf.de/

MHCPEP（MHC结合肽数据库）

http://wehih.wehi.edu.au/mhcpep/

小鼠基因组数据库

http://www.informatics.jax.org/mgd.html

小鼠基因敲除数据库（Mouse Knockout Database）

http://BioMedNet.com/cgi-bin/mko/mkobrwse.pl

ATCC（美国菌种保藏中心）

http://www.atcc.org/

高度保守的核蛋白序列的组蛋白序列数据库

http://www.ncbi.nlm.nih.gov/Baxevani/HISTONES

3Dee（蛋白质结构域定义数据库）

http://barton.ebi.ac.uk/servers/3Dee.html

InterPro（蛋白质域以及功能位点的完整资源）

http://www.ebi.ac.uk/interpro/

NRL_3D（由PDB数据库、图片以及搜索得到的序列结构数据库）

http://www.gdb.org/Dan/proteins/nr13d.html

VBASE人类可变免疫基因序列

http://www.mrc-cpe.cam.ac.uk/imt-doc/public/INTRO.html

GPCRD（G蛋白结合受体数据）

http://www.gpcr.org/7tm/

人类细胞遗传学（染色体及染色体组学）

http://www.selu.com/bio/cyto/human/index.html

蛋白激酶资源

http://www.sdsc.edu/projects/Kinases/pkr/pk_info.html#Format

碳水化合物数据库

http://www.boc.chem.ruu.nl/sugabase/databases.html

包柔氏螺旋体菌分子生物学主页

http://www.pasteur.fr/Bio/borrelia/Welcome.html

人类乳头瘤病毒数据库

http://HPV-web.lanl.gov/

用于人类健康与疾病的蛋白质组分析的二维电泳数据库

http://biobase.dk/cgi-bin/celis

DBA哺乳动物基因组大小数据库

http://www.unipv.it/~webbio/dbagsh.htm

DOGS（基因组规模数据库）

http://www.cbs.dtu.dk/databases/DOGS/index.html

美国专利引用数据库

http://cos.gdb.org/repos/pat/

13.5 序列相似性搜索

EBI序列相似性研究网页

http://www.ebi.ac.uk/searches/searches.html

NCBI：BLAST注释

http://www.ncbi.nlm.nih.gov/BLAST/

EMBL的BLITZ ULTRA快速搜索

http://www.ebi.ac.uk/searches/blitz_input.html

EMBL WWW服务器

　　http://www.embl-heidelberg.de/Services/index.html#5

蛋白质或核苷酸的模式浏览

　　http://www.mcs.anl.gov/compbio/PatScan/HTML/patscan.html

MEME（蛋白质超二级结构模体发现与研究）

　　http://meme.sdsc.edu/meme/website/

CoreSearch（DNA序列保守元件的识别）

　　http://www.gsf.de/biodv/coresearch.html

PRINTS/PROSITE浏览（搜索motif数据库）

　　http://www.biochem.ucl.ac.uk/cgi-bin/attwood/SearchPrintsForm.pl

苏黎世ETH服务器的DARWIN系统

　　http://cbrg.inf.ethz.ch/

利用动态规划找出序列相似性的Pima Ⅱ

　　http://bmerc-www.bu.edu/protein-seq/pimaⅡ-new.html

利用与模式库进行哈希码（hashcode）比较找到序列相似性的DashPat

　　http://bmerc-www.bu.edu/protein-seq/dashPat-new.html

PROPSEARCH（基于氨基酸组成的搜索，EMBL）

　　http://www.embl-heidelberg.de/aaa.html

序列搜索协议（集成模式搜索）

　　http://www.biochem.ucl.ac.uk/bsm/dbbrowser/protocol.html

ProtoMap（SWISS-PROT中所有蛋白质的自动层次分类）

　　http://www.protomap. cs.huji.ac.il/

GenQuest（利用Fasta、Blast、Smith-Waterman方法在任意数据库中搜索）

　　http://www.gdb.org/Dan/gq/gq.form.html

SSearch（对特定数据库的搜索）

　　http://watson.genes.nig.ac.jp/homology/ssearch-e_help.html

Peer Bork搜索列表（motif/模式/序列谱搜索）

　　http://www.embl-heidelberg.de/~bork/pattern.html

PROSITE数据库搜索（搜索序列的功能位点）

　　http://www.ebi.ac.uk/searches/prosite.html

PROWL（Skirball研究中心的蛋白质信息检索）

　　http://mcphar04.med.nyu.edu/index.html

CEPH基因型数据库

http://www.cephb.fr/cephdb/

13.6 比 对

13.6.1 序列和结构的两两比对

蛋白质两两比对（SIM）

http://expasy.hcuge.ch/sprot/sim-prot.html

LALNVIEW比对可视化观察程序

ftp://expasy.hcuge.ch/pub/lalnview

BCM搜索装置（两两序列比对）

http://searchlauncher.bcm.tmc.edu/seq-search/alignment.html

DALI蛋白质三维结构比较

http://www2.ebi.ac.uk/dali/

DIALIGN（无间隙罚分的比对程序）

http://www.gsf.de/biodv/dialign.html

13.6.2 多重序列比对及系统进化树

ClustalW（BCM的多重序列比对）

http://searchlauncher.bcm.tmc.edu/multi-align/multi-align.html

PHYLIP（推测系统进化树的程序）

http://evolution.genetics.washington.edu/phylip.html

其他系统进化树程序，PHYLIP文档的汇编

http://expasy.hcuge.ch/info/phylogen.sof

生命树（tree of life）主页（系统进化树及生物多样性的相关信息）

http://phylogeny.arizona.edu/tree/phylogeny.html

向古植物学家提供的链接

http://www.uni-wuerzburg.de/mineralogie/palbot1.html

系统进化树分析程序（生命树列表）

http://phylogeny.arizona.edu/tree/programs/programs.html

遗传分类学

http://www.kheper.auz.com/gaia/biosphere/systematics/cladistics.htm

遗传分类学软件（Willi Hennig协会提供的列表）

http://www.cladistics.org/education.html

用于多重序列比对的BCM搜索装置

http://searchlauncher.bcm.tmc.edu/multi-align/multi-align.html

AMAS（分析多重序列比对中的序列）

http://barton.ebi.ac.uk/servers/amas_server.html

维也纳RNA二级结构软件包

http://www.tbi.univie.ac.at/~ivo/RNA/

WebLogo（序列标识）

http://www.bio.cam.ac.uk/cgi-bin/seqlogo/logo.cgi

使用相对熵的蛋白质序列标识

http://www.cbs.dtu.dk/gorodkin/appl/plogo.html

RNA结构序列标识

http://www.cbs.dtu.dk/gorodkin/appl/slogo.html

RNA互信息图

http://www.gorodkin/appl/MatrixPlot/mutRNA/

13.7　有代表性的预测服务器

13.7.1　从序列预测蛋白质结构

PHD蛋白质预测服务器，用于二级结构、水溶性以及跨膜片断的预测

http://www.embl-heidelberg.de/predictprotein/predictprotein.html

PhdThreader（利用逆折叠方法预测、识别折叠类）

http://www.embl-heidelberg.de/predictprotein/phd_help.html

PSIpred（蛋白质结构预测服务器）

http://insulin.brunel.ac.uk/psipred/

THREADER（戴维・琼斯）

http://www.biochem.ucl.ac.uk/~jones/threader.html

TMHMM（跨膜螺旋蛋白的预测）

http://www.cbs.dtu.dk/services/TMHMM/

蛋白质结构分析，BMERC

http://bmerc-www.bu.edu/protein-seq/protein-struct.html

蛋白质域和折叠类预测的提交表

http://genome.dkfz-heidelberg.de/nnga/def-query.html

NNSSP（利用最近相邻法预测蛋白质的二级结构）

http://genomic.sanger.ac.uk/pss/pss.html

Swiss-Model（基于知识的蛋白质自动同源建模服务器）

http://www.expasy.ch/swissmod/SWISS-MODEL.html

SSPRED（用多重序列比对进行二级结构预测）

http://www.mrc-cpe.cam.ac.uk/jong/predict/sspred.htm

SSCP（满足氨基酸组成约束的二级结构预测）

http://www.mrc-cpe.cam.ac.uk/jong/predict/sscp.htm

法国IBCP的SOPM（自寻优化预测方法、二级结构）

http://pbil.ibcp.fr/cgi-bin/npsa_automat.pl?page=/NPSA/npsa_sopm.html

NNPREDICT（利用神经网络进行残基的预测）

http://www.cmpharm.ucsf.edu/~nomi/nnpredict.html

SSpro（三类别的二级结构）

http://promoter.ics.uci.edu/BRNN-PRED/

SSpro8（八类别的二级结构）

http://promoter.ics.uci.edu/BRNN-PRED/

ACCpro（水溶性）

http://promoter.ics.uci.edu/BRNN-PRED/

CONpro（相邻残基的数目）

http://promoter.ics.uci.edu/BRNN-PRED/

TMAP（蛋白质跨膜片断的预测服务）

http://www.embl-heidelberg.de/tmap/tmap_info.html

TMpred（跨膜区域和方向的预测）

http://www.ch.embnet.org/software/TMPRED_form.html

MultPredict（多重序列比对的序列的二级结构）

http://kestrel.ludwig.ucl.ac.uk/zpred.html

NIH分子建模主页（带有链接的建模主页）

http://cmm.info.nih.gov/modeling/

BCM搜索装置（蛋白质二级结构预测）

http://searchlauncher.bcm.tmc.edu/seq-search/struc-predict.html

COILS（蛋白质的卷曲螺旋区域预测）

http://www.ch.embnet.org/software/coils/COILS_doc.html

Coiled Coils（卷曲螺旋）

http://www.york.ac.uk/depts/biol/units/coils/coilcoil.html

Paircoil（氨基酸序列中的卷曲螺旋定位）

http://theory.lcs.mit.edu/bab/webcoil.html

PREDATOR（由单序列预测蛋白质二级结构）

http://www.embl-heidelberg.de/argos/predator/predator_info.html

DAS（Dense Alignment Surface，密度比对表面和蛋白质跨膜区域的预测）

http://www.biokemi.su.se/~server/DAS/

UCLA-DOE结构预测服务器的折叠识别

http://www.doe-mbi.ucla.edu/people/frsvr/frsvr.html

分子建模服务器及数据库

http://bionmr5.bham.ac.uk/modelling/model.html

EVA（蛋白质结构预测服务器的自动评估）

http://cubic.bioc.columbia.edu/eva/

13.7.2 基因发现与内含子剪接位点识别

NetGene（人类基因内含子剪接位点预测）

http://www.cbs.dtu.dk/services/NetGene2/

NetPlantGene（阿布属拟南芥的内含子剪接位点预测）

http://www.cbs.dtu.dk/services/NetPGene

GeneQuiz（基因组自动分析）

http://www.sander.embl-heidelberg.de/genequiz/

GRAIL界面（蛋白质编码区域与功能位点）

http://avalon.epm.ornl.gov/Grail-bin/EmptyGrailForm

GENEMARK（蛋白质编码区域预测的WWW系统）

http://genemark.biology.gatech.edu/GeneMark

GENSCAN网络服务器：基因组DNA的完整基因结构

http://gnomic.stanford.edu/~chris/GENSCANW.html

FGENEH Genefinder（人类DNA序列的基因结构预测）

http://mbcr.bcm.tmc.edu/Guide/Genefinder/fgeneh.html

GRAIL和GENQUEST（通过E-mail进行序列分析、基因拼接和序列比较）

http://avalon.epm.ornl.gov/manuals/grail-genquest.9407.html

CpG岛发现程序

http://www.ebi.ac.uk/cpg/

真核细胞Pol Ⅱ启动子预测

http://biosci.umn.edu/software/proscan.html

启动子预测输入

http://www-hgc.lbl.gov/projects/promoter.html

网络信号扫描服务器（浏览DNA序列以找到真核细胞的转录元件）

http://bimas.dcrt.nih.gov/molbio/signal/

基因发现网页

http://konops.imbb.forth.gr/~topalis/mirror/gdp.html

基因组测序计划列表

http://www.mcs.anl.gov/home/gaasterl/genomes.html

13.7.3 DNA微阵列数据和方法

Cyber-T（DNA微阵列数据分析服务器）

http://128.200.5.223/CyberT/

布朗实验室（Brown Lab）的微阵列指南

http://cmgm.stanford.edu/pbrown

斯坦福微阵列数据库

http://genome-www4.stanford.edu/MicroArray/SMD/

斯坦福微阵列论坛

http://cmgm.stanford.edu/cgi-bin/cgiwrap/taebshin/dcforum/dcboard.cgi

EBI的Brazma微阵列网页

http://industry.ebi.ac.uk/~brazma/Data-mining/microarray.html

基因表达和DNA微阵列技术的网络资源

http://industry.ebi.ac.uk/~alan/MicroArray/

Gene-X（阵列数据管理和分析系统）

http://www.ncgr.org/research/genex/

UCI功能基因组阵列工具和软件

http://www.genomics.uci.edu/

Matern的DNA微阵列网页

　　http://barinth.tripod.com/chips.html

微阵列信息、工具和协议的公共资源

　　http://www.microarrays.org/

Weisshaar的DNA微阵列链接列表

　　http://www.mpiz-koeln.mpg.de/~weisshaa/Adis/DNA-array-links.html

用DNA微阵列技术识别基因控制精子形成的过程

　　http://www.mcb.arizona.edu/wardlab/microarray.html

13.7.4　其他预测服务器

NetStart（脊椎动物和阿布属拟南芥DNA的翻译起始）

　　http://www.cbs.dtu.dk/services/NetStart/

NetOGlyc（哺乳动物蛋白质O联糖基化位点）

　　http://www. cbs.dtu.dk/services/NetOGlyc/

YinOYang（真核生物蛋白质序列的O-β-GlcNAc位点）

　　http://www.cbs.dtu.dk/services/YinOYang/

SignalP

　　（革兰氏阳性菌、革兰氏阴性菌和真核生物蛋白质的信号肽及剪切位点）

　　http://www.cbs.dtu.dk/services/SignalP/

NetChop（人类蛋白酶体的剪切位点）

　　http://www.cbs.dtu.dk/services/NetChop/

NetPhos（真核细胞蛋白质的丝氨酸、苏氨酸及酪氨酸磷酸化作用位点）

　　http://www.cbs.dtu.dk/services/NetPhos/

TargetP（亚细胞位置预测）

　　http://www.cbs.dtu.dk/services/TargetP/

ChloroP（叶绿体分选信号预测）

　　http://www.cbs.dtu.dk/services/SignalP/

PSORT（由序列预测蛋白质分选信号及序列定位）

　　http://psort.nibb.ac.jp/

PEDANT（蛋白质提取、描述及分析工具）

　　http://pedant.mips.biochem.mpg.de/

将提交的序列与COG数据库里的序列进行比较

　　http://www.ncbi.nlm.nih.gov/COG/cognitor.html

从序列预测HLA结合肽

http://www-bimas.dcrt.nih.gov/molbio/hla_bind/index.html

13.8 分子生物学软件链接

生物信息学可视化工具

http://industry.ebi.ac.uk/alan/VisSupp/

EBI分子生物学软件档案

http://www.ebi.ac.uk/software/software.html

BioCatalog

http://www.ebi.ac.uk/biocat/e-mail_Server_ANALYSIS.html

生物学软件和数据库档案

http://www.gdb.org/Dan/softsearch/biol-links.html

巴顿研究组（Barton group）的软件

（ALSCRIPT、AMPS、AMAS、STAMP、ASSP、JNET和SCANPS）

http://barton.ebi.ac.uk/new/software.html

科恩研究组（Cohen group）的软件

（旋转异构体库、BLoop、QPack、FOLD和Match）

http://www.cmpharm.ucsf.edu/cohen/pub/

沃兹沃思中心（Wadsworth Center）的贝叶斯生物信息学

http://www.wadsworth.org/res&res/bioinfo/

Rasmol软件和脚本文件

http://scop.mrc-lmb.cam.ac.uk/std/rs/

MolScript

http://indl.mrc-lmb.cam.ac.uk/external-file-copies/molscript.html

WHAT IF

http://www.hgmp.mrc.ac.uk/Registered/Option/whatif.html

Biosym（Discover）

http://ind1.mrc-lmb.cam.ac.uk/external-file-copies/biosym/discover/html/Disco_Home.html

UC Santa Cruz的序列保守性HMM的SAM软件

http://www.cse.ucsc.edu/research/compbio/sam.html

HMMER（隐马氏模型软件的源代码）

http://hmmer.wustl.edu/

ClustalW

http://www.ebi.ac.uk/clustalw/

DSSP程序

http://www.sander.embl-heidelberg.de/dssp/

用于病毒重组的Bootscanning

http://www.bio.net//hypermail/RECOMBINATION/recom.199607/0004.html

用于大家系连锁分析的块状吉布斯采样

http://www.cs.auc.dk/~claus/block.html

ProMSED（用于Windows的蛋白质多序列编辑器）

ftp://ftp.ebi.ac.uk/pub/software/dos/promsed/

用于Sun/Solaris的DBWatcher

http://www-igbmc.u-strasbg.fr/Biolnfo/LocalDoc/DBWatcher/

ProFit（蛋白质最小二乘拟合软件）

http://www.bioinf.org.uk/software/

印第安纳大学IUBIO软件和数据

http://iubio.bio.indiana.edu/

NIH分子生物学软件列表

http://bimas.dcrt.nih.gov/sw.html

用于蛋白质/肽分析的ProAnalyst软件

ftp://ftp.ebi.ac.uk/pub/software/dos/proanalyst/

使用距离几何的DRAGON蛋白质建模工具

http://www.nimr.mrc.ac.uk/~mathbio/a-aszodi/dragon.html

分子界面软件包

http://www.best.com/~connolly/

生物技术软件和因特网期刊

http://www.orst.edu/~ahernk/bsj.html

MCell（细胞微生理学的蒙特卡罗仿真）

http://www.mcell.cnl.salk.edu/

HHMpro（使用图形界面进行序列分析的HMM仿真）

http://www.netid.com/ html/hmmpro.html

13.9 网上的博士课程

生物计算课程资源列表：课程大纲

http://www.techfak.uni-bielefeld.de/bcd/Curric/syllabi.html

生物序列分析和蛋白质建模的Ph.D.课程

http://www.cbs.dtu.dk/phdcourse/programme.html

分子科学虚拟学校

http://www.ccc.nottingham.ac.uk/vsms/sbdd/

EMBnet生物计算指南

http://biobase.dk/Embnetut/Universl/embnettu.html

蛋白质结构的合作课程

http://www.cryst.bbk.ac.uk/PPS/index.html

自然科学GNA虚拟学校

http://www.techfak.uni-bielefeld.de/bcd/Vsns/index.html

分子生物学算法

http://www.cs.washington.edu/education/courses/590bi/

ISCB教育工作组

http://www.sdsc.edu/pb/iscb/iscb-edu.html

13.10 生物信息学协会

国际计算生物学协会（ISCB）

http://www.iscb.org/

北欧国家生物信息学协会

http://www.socbin.org/

日本生物信息学协会

http://www.jsbi.org/

13.11 HMM/NN仿真软件

本书中所描述的大量研究实例使用Net-ID有限公司和哥本哈根的丹麦生物序列分析中心的研究人员合作开发的用于生物序列分析的机器学习软件环境加以实现。

这一软件环境的基础是Net-Libs，这是由Net-ID开发的用于图形建模、机器学习和推理的面向对象的C++类库。这个库支持任何图模型（NN、HMM、贝叶斯网络等）的分层及递归实现，以及在推断/学习过程和动态规划中传递信息、错误和常用资源的局部信息传递算法。

除此以外，Net-Libs还为将HMM仿真和NN仿真用于生物序列分析提供了基础。它运用Java语言实现了简便易用的图形界面。该软件可在Unix和NT平台下运行。

另外，该软件环境还可方便地进行输入/输出序列、数据库、文件以及训练模型库的操作。其中，训练模型库包括了大量蛋白质家族和DNA元件（启动子、剪接位点和外显子等）的HMM模型，以及在蛋白质和DNA序列中检测特殊结构或功能信号的大量NN模型。

若要得到更多信息，请联系：admin@netid.com.



附录A　统计学

A.1　决策理论和损失函数

在任何决策问题中，[238,63,431] 人们必须定义一个损失函数（loss function）［或等价的一个回报函数（reward function）］来度量一定环境下采取一定行动所产生的效果。决策理论的基本原理是：用一小组合适的公理描述理性行为，在这组公理下，最优策略是能够最小化损失的期望值的一种策略。其中，期望根据现有知识通过对不确定环境的贝叶斯概率分析加以定义。注意，一些纯粹的科学数据分析工作——例如数据压缩、重构或聚类——本质上都是决策理论问题，因此需要定义一个损失函数。甚至连预测也可以被归结为这一类问题，这也就是为什么在回归中，当损失函数是二次函数时，$\mathbf{E}(y|x)$ 是在给定x的条件下y的最优估计（见下文）。

如果我们的目标是找出“最优”模型（这种情况在本书里经常出现），期望的损失函数就是负对数似然（或负对数先验）函数。但是一般来讲，这两个函数是有区别的。例如从原理上说，高斯数据也有二次负对数似然度，但人们却使用一个二次损失函数。

可以根据最小化特性定义两个损失函数 f_1 和 f_2 的等价性。等价的条件是，存在一个阶数不变的变换g［若$u \leqslant v$，则$g(u) \leqslant g(v)$］使得 $f_2=gf_1$。在这种条件下，f_1 和 f_2 有相同的极小点。这当然并不意味着用于 f_1 和 f_2 的最小化（即学习）算法会表现出相同的方式，也不意味着 f_1 和 f_2 在它们的极小值点附近有相同的曲率。正如在第5章提到的，当 $\sum p_i=1$时，二次函数 $f_1(y)=\sum_1^K(p_i-y_i)^2/2$和交叉

熵函数$f_2(y)=-\sum_1^K p_i \log y_i$提供了一个很好的例子。如果 f_2 满足 $\sum y_i=1$，则以上两个函数都是y的凸函数，且在$y_i=p_i$处有惟一的全局极小值。事实上，把 f_2 在p_i处泰勒展开，有

$$f_2(y)=-\sum_1^K p_i \log(p_i+\varepsilon_i)\approx \mathcal{H}(p)+\sum_1^K \frac{\varepsilon_i^2}{2p_i} \tag{A.1}$$

其中$y_i=p_i+\varepsilon_i$，$\sum\varepsilon_i=0$。因此，当$p_i=1/K$是均匀分布的时候，甚至有更强的结果$f_2\approx\mathcal{H}(p)+Kf_1$。这样，去掉常数项，二次损失函数和交叉熵损失函数 f_1 和 f_2 在相同的最优点附近重合且有相同的曲率。在这个附录的余下部分，将集中讨论最常用的二次损失函数（或高斯似然函数）。但是利用上面的讨论，很多结果可以推广到其他损失函数。

A.2 二次损失函数

A.2.1 基本分解

我们先考虑一串数$y_1,\cdots,y_K$和二次型 $f(y)=\sum_1^K(y-y_i)^2/K$，即均方损失。此时 f 在平均值 $y^*=\mathbf{E}(y)=\sum_1^K y_i/K$处有惟一极小值。利用Jensen不等式（附录B）可以很容易地看出这一点，写得更直接一些有

$$\begin{aligned}f(y)&=\frac{1}{K}\sum_1^K\left(y-y^*+y^*-y_i\right)^2\\&=\left(y-y^*\right)^2+\frac{1}{K}\sum_1^K\left(y^*-y_i\right)^2+\frac{2}{K}\sum_1^K\left(y-y^*\right)\left(y^*-y_i\right)\\&=\left(y-y^*\right)^2+\frac{1}{K}\sum_1^K\left(y^*-y_i\right)^2\geqslant f\left(y^*\right)\end{aligned} \tag{A.2}$$

因此 f 可以被分解为偏差 $\left(y-y^*\right)^2$ 和方差 $\sum_1^K\left(y^*-y_i\right)^2$ 的和。偏差度量的是从y到最优均值的距离，方差度量的是y_i在均值附近的离散度。这个将二次损失函数转化为两个二次项之和的分解（毕达哥拉斯定理）很重要，在该分解中所有的交叉项都被消掉了。该分解的各种不同变形下面将会反复使用。当y_i以不同频率或

强度$p_i \geq 0$出现且$\sum p_i = 1$时，上述结论总是成立。这里，期望的二次损失在加权平均$y^* = \mathbf{E}(y) = \sum p_i y_i$处取最小值，因为存在如下分解

$$\mathbf{E}\left[(y-y_i)^2\right] = \sum_1^K p_i (y-y_i)^2 = (y-y^*)^2 + \sum_1^K p_i (y^*-y_i)^2 \tag{A.3}$$

现在说明如何将这一简单的分解用于回归问题。具体的做法是在几个方向上用一些略微不同的期望算子，包括对不同的训练集或不同的估计器求平均。

A.2.2 回归上的应用

考虑这样一个回归问题：x和y由分布$P(x, y)$刻画，而我们试图利用数据x和y来估计目标函数$f(x)$。与第5章一样，为简化起见，我们假设由于存在噪声，对于一个x可能存在不同的y与之对应。对于任意x，期望误差或损失$\mathbf{E}[(y-f(x))^2|x]$的最小值点是条件期望$y^* = \mathbf{E}(y|x)$，这里所有的期望值都是对分布P求取的，也可以用对应的样本来近似。写成如下形式后可将平方式展开：

$$\mathbf{E}\left[(y-f(x))^2|x\right] = \mathbf{E}\left[(y-\mathbf{E}(y|x)+\mathbf{E}(y|x)-f(x))^2|x\right] \tag{A.4}$$

很容易看出交叉项消失了，只剩下偏差和方差两部分：

$$\mathbf{E}\left[(y-f(x))^2|x\right] = \left[\mathbf{E}(y|x)-f(x)\right]^2 + \mathbf{E}\left[(y-\mathbf{E}(y|x))^2|x\right] \tag{A.5}$$

A.3 偏差/方差均衡

考虑同样的回归体系，但是使用不同的训练集D。对每一个训练集D，学习算法产生一个不同的估计$f(x, D)$。这样一个估计器的性能可以用期望损失$\mathbf{E}[(y-f(x,D))^2|x, D]$来度量，这里期望仍然是对$P$求取的。通过一般的计算可得

$$\mathbf{E}\left[(y-f(x,D))^2|x,D\right] = \left[f(x,D)-\mathbf{E}(y|x)\right]^2 + \mathbf{E}\left[(y-\mathbf{E}(y|x))^2|x,D\right] \tag{A.6}$$

其中，方差项不依赖于训练样本D。因此，对任意x，估计器$f(x, D)$的有效性由偏差$[f(x, D)-\mathbf{E}(y|x)]^2$来度量，也就是说，由它偏离最优估计$\mathbf{E}(y|x)$的程度来度量。现在来看这种误差对所有给定大小的训练集D的平均。记

$$\mathbf{E}_D\left[\left(f(x,D)-\mathbf{E}(y|x)\right)^2\right]=$$
$$\mathbf{E}_D\left[\left(f(x,D)-\mathbf{E}_D(f(x,D))+\mathbf{E}_D(f(x,D))-\mathbf{E}(y|x)\right)^2\right] \quad \text{(A.7)}$$

消掉交叉项，剩下偏差-方差分解

$$\mathbf{E}_D\left[\left(f(x,D)-\mathbf{E}(y|x)\right)^2\right]=$$
$$\left[\mathbf{E}_D(f(x,D))-\mathbf{E}(y|x)\right]^2+\mathbf{E}_D\left[\left(f(x,D)-\mathbf{E}_D(f(x,D))\right)^2\right] \quad \text{(A.8)}$$

偏差-方差分解对应着机器学习里的一种不确定原理：试图减少一项而不同时增加另一项，往往是很困难的。这也是在数据的欠拟合和过拟合之间的基本平衡。一个具有大量参数的学习机较灵活，可以覆盖很大的函数空间，达到很小的偏差。但是，这种学习机对数据很敏感，因此与数据的过拟合关系密切，方差因此将趋于很大。一个简单的学习机一般有较小的方差，但代价是有较大的欠拟合偏差。

A.4 估计器的组合

正如在第4章中提到的，用一个与估计器关联的参数w的离散（甚至是连续）分布$p_w \geqslant 0$（$\sum_w p_w=1$）把不同的估计器$f(x,w)$组合起来，有时会很有用。例如（A.8）中不同的估计器可以对应不同的训练集。通过对w求期望，（A.8）马上可以推广成

$$\mathbf{E}_w\left[\left(f(x,w)-\mathbf{E}(y|x)\right)^2\right]=$$
$$\left[\mathbf{E}_w\left(f(x,w)-\mathbf{E}(y|x)\right)\right]^2+\mathbf{E}_w\left[\left(f(x,w)-\mathbf{E}_w(f(x,w))\right)^2\right] \quad \text{(A.9)}$$

因此加权平均估计器的损失（有时也称做集合平均）$f^*(x)=\mathbf{E}_w(f(x,w))$总是比平均损失小：

$$\mathbf{E}_w\left[\left(f(x,w)-\mathbf{E}(y|x)\right)^2\right]\geqslant\left[f^*(x)-\mathbf{E}(y|x)\right]^2 \quad \text{(A.10)}$$

事实上，我们可以利用分布P求（A.9）对所有x的平均，从而得到“推广”误差：

$$\mathbf{E}_X\left[f^*(x)-\mathbf{E}(y|x)\right]^2=$$

$$\mathbf{E}_X\mathbf{E}_w\left[\left(f(x,w)-\mathbf{E}(y|x)\right)^2\right]-\mathbf{E}_X\mathbf{E}_w\left[\left(f(x,w)-f^*(x)\right)^2\right] \tag{A.11}$$

这就是参考文献［340，339］中用的关系式。左边是集合的期望损失。右边的第一项是对估计器的期望损失，第二项被称做模糊度（ambiguity）。显然，将同样的估计器结合起来是没有用的。因此这个模型集方法的一个有用的必要条件是每个估计器显著不同，或者说模糊度应该很大。做到这一点的一种方法是对每个估计器用不同的训练集（见参考文献［340］，其中还讨论了求最优加权方案p_w算法——例如使用二次规划方法）。很重要的一点是估计器之间的所有关联都包含在模糊度这一项里了。模糊度这一项不依赖于任何目标值，因此可以用没有标明类别的数据估计出来。

A.5 误差带

考虑在带有一个参数w和一个均匀先验分布的情况下建立模型。令$f(w) = -\log \mathbf{P}(D|w)$为数据的负对数似然函数。在不是很严格的可导性条件下，最大似然估计w^*满足$f'(w^*)=0$。因此，在w^*的邻域里，我们可以将$f(w^*)$展开成泰勒级数：

$$f(w) \approx f(w^*)+\frac{1}{2}f''(w^*)(w-w^*)^2 \tag{A.12}$$

或

$$\mathbf{P}(D|w)=e^{-f(w)} \approx Ce^{-\frac{1}{2}f''(w^*)(w-w^*)^2} \tag{A.13}$$

这里$C=e^{-f(w^*)}$。因此似然函数和后验分布$\mathbf{P}(w|M)$在局部上类似于一个标准差为$1/\sqrt{f''(w^*)}$的高斯分布，曲率为f。在多维的情况下，2阶偏导数矩阵叫做Hessian阵。因而，对数似然函数的Hessian阵有一个几何上的解释，它在很多问题中扮演着重要的角色。这也叫做Fisher信息量矩阵（见参考文献［5，16，373］）。

A.6 充分统计量

很多统计问题可以通过充分统计量来简化。一个参数w的充分统计量是数据

的一个函数，该函数包含了数据中所有与w有关的信息。更正式地，考虑一个随机变量X，其分布具有参数w。如果条件分布$P(X=x|S(X)=s)$以概率1与w独立，其中S为X的一个函数，则S是w的一个充分统计量。因此$P(X=x|S(X)=s)$不随w改变，或者说

$$\mathbf{P}\left(X=x|S=s,w\right)=\mathbf{P}\left(X=x|S=s\right) \tag{A.14}$$

如果我们用任意一个统计量$H=h(X)$代替X，以上等式仍然成立。等价地，可以由这个等式得到$\mathbf{P}(w|X,S)=\mathbf{P}(w|S)$。所有关于$w$的信息都由$S$包含了，任何其他的统计量都是多余的。特别是充分统计量包含了互信息I（见附录B）：$I(w,X)=I(w,S(X))$。

作为一个例子，考虑从随机变量$\mathcal{N}(\mu,\sigma^2)$中抽取出一个样本$X=(X_1,\cdots,X_N)$，其中$w=(\mu,\sigma)$。那么$(m,s)$是$w$的一个充分统计量，其中$m=\sum_i X_i/N$，$s^2=\sum_i(X_i-m)^2/(N-1)$。换句话说，样本中所有有关$\mu$的信息都被包含在样本均值$m$中，而所有有关方差的信息都被包含在$s^2$中。

A.7 指数族

指数族[94]是一个最重要的概率分布族。它的应用范围广泛并具有独特的计算特性：它的各种不同形式是很多数据分析快速算法的核心。很多统计上的一般定理可以在这个独特的参数化分布族上得到证明。单参数指数族具有如下形式的密度函数：

$$\mathbf{P}(x|w)=c(w)h(x)e^{q(w)S(x)} \tag{A.15}$$

大多数常用分布属于指数族，包括正态分布（均值或方差固定）、χ^2分布、二项分布和多项分布、几何分布和负二项分布、指数和伽玛分布、贝塔分布、泊松分布和Dirichlet分布。本书用到的所有分布都属于指数族。指数族有一个重要性质，即从单参数指数族的一个分布中随机抽取的一个样本总是具有一个充分统计量S。进一步，充分统计量本身具有的分布也属于指数族。

A.8 其他有用分布

这里我们简单回顾一下在第12章中用到的另外三个连续分布。

A.8.1 标定的逆伽玛分布

自由度$\nu>0$，尺度因子为$s>0$的标定逆伽玛分布$\mathcal{I}(x;\nu,s^2)$由下式给出：

$$\frac{(\nu/2)^{\nu/2}}{\Gamma(\nu/2)} s^{\nu} x^{-(\nu/2+1)} e^{-\nu s^2/(2x)} \tag{A.16}$$

其中$x>0$。当$\nu>2$时期望为（$\nu/\nu-2$）s^2，否则期望为无穷。众数总是（$\nu/\nu+2$）s^2。

A.8.2 学生氏分布

自由度$\nu>0$，位置为m，尺度$\sigma>0$的学生氏-t分布$t(x;\nu,m,\sigma^2)$由下式给出：

$$\frac{\Gamma((\nu+1)/2)}{\Gamma(\nu/2)\sqrt{\nu\pi}\sigma}\left(1+\frac{1}{\nu}\left(\frac{x-m}{\sigma}\right)^2\right)^{-(\nu+1)/2} \tag{A.17}$$

它的均值和众数都为m。

A.8.3 逆Wishart分布

逆Wishart分布$\mathcal{I}(W;\nu,S^{-1})$由下式给出：

$$\left(2^{\nu k/2}\pi^{k(k-1)/4}\prod_{i=1}^{k}\Gamma\left(\frac{\nu+1-i}{2}\right)\right)^{-1}|S|^{\nu/2}|W|^{-(\nu+k+1)/2}$$
$$\exp\left(-\frac{1}{2}tr\left(SW^{-1}\right)\right) \tag{A.18}$$

其中ν代表自由度，S是一个$k\times k$的对称正定尺度矩阵。W也是正定的，W的期望是$E(W)=(\nu-k-1)^{-1}S$。

A.9 变分法

为了理解这一部分，读者必须熟悉相对熵的概念（附录B）。在贝叶斯体系中，我们经常面临一些难以处理的高维概率分布$P(x)=P(x_1,\cdots,x_n)$，它们过于复杂以至于难以精确估计。变分法（variational method）的基本思想是构造一个易于处理的带有参数θ的分布族$Q(x,\theta)$，通过选取这个分布族中离P最近的那个分布来近似$P(x)$。这需要有一种度量概率分布之间距离的方法。在变分法中，通常用相对熵或KL距离$\mathcal{H}(Q,P)$来做这件事。因此我们试图最小化

$$\mathcal{H}(Q,P)=\sum Q\log\frac{Q}{P}=-\mathcal{H}(Q)+\mathbf{E}_Q(-\log P) \tag{A.19}$$

如果用波耳兹曼-吉布斯分布$P=e^{-\lambda\mathcal{E}}/Z(\lambda)$表示$P$，则

$$\mathcal{H}(Q,P)=-\mathcal{H}(Q)+\lambda\mathbf{E}_Q(\mathcal{E})+\log Z(\lambda)=\lambda\mathcal{F}+\log Z(\lambda) \quad (A.20)$$

其中$\mathcal{F}$是第3章中定义的自由能。由于分割函数Z不依赖于θ，因此最小化$\mathcal{H}$就等价于最小化$\mathcal{F}$。由附录B中的Jensen不等式，我们知道，对于任何近似的Q，存在$\mathcal{H}\geqslant 0$，或等价地存在$\mathcal{F}\geqslant -\log Z(\lambda)/\lambda$。在最优点的等号仅当$Q^*=P$时成立。

在建模时，我们经常有一族参数化的模型，其中参数为w，P是后验分布$\mathbf{P}(w|D)$。利用贝叶斯理论和前面的方程，我们可以得到

$$\mathcal{H}(Q,P)=-\mathcal{H}(Q)+\mathbf{E}_Q[-\log\mathbf{P}(D|w)-\log\mathbf{P}(w)]+\log\mathbf{P}(D) \quad (A.21)$$

其中$\lambda=1,\mathcal{E}=-\log\mathbf{P}(D|w)-\log\mathbf{P}(w)$。近似分布仍然必须满足$\mathcal{H}\geqslant 0$或$\mathcal{F}\geqslant -\log\mathbf{P}(D)$。

在某种意义上，变分法与高层次的贝叶斯推断相近，因为它们都试图近似整个分布$\mathbf{P}(w|D)$，而不像MAP估计那样仅仅关心众数。在一个更高的层次上，我们可以看整个Q空间的分布而不仅仅是最优分布Q^*。作为练习，读者可以进一步研究变分法在贝叶斯体系中的位置，还可以问一问自己，变分法本身能否看做MAP估计的一种形式。

当然变分法的根本问题还是近似函数族$Q(x,\theta)$或$Q(w,\theta)$的选取问题。这个函数族必须满足两个相互冲突的条件：它必须足够简单以便于计算，但它又不能太简单，否则距离$\mathcal{H}(Q,P)$就太大了。易于计算指的是，人们应该能够估计诸如$\mathcal{F}$和$\partial\mathcal{F}/\partial\theta$之类的量。一个简单的情形是，函数族$Q$是因子分布。$Q$是一个因子分布，当且仅当它有如下函数形式$Q(x_1,\cdots,x_n)=Q(x_1)\cdots Q(x_n)$。统计力学的均值场理论是带有因子近似的变分法的一个特例（见参考文献[582]）。更一般地，如何构造合适的近似函数族Q取决于具体问题，它更像一门艺术而不仅仅是科学。然而在构造Q时，下面几点很有用：

- 混合分布
- 指数族分布
- 独立性假设和相应的因子化（附录C）

例如，Q可以写做因子分布的混合分布，这里每个因子都属于指数族。需要优化的参数是每个指数分量的混合系数和/或参数（均值、方差）。

附录B 信息论、熵和相对熵

在这里我们简单地总结一下在本书和其他很多机器学习问题中用到的信息论的最基本概念，更深入的内容见参考文献［483，71，137，577］。信息的三个最基本概念和量是熵、互信息和相对熵。在研究信息如何通过编码、传输和压缩等各种操作发生转换的问题时，这些概念极为重要。相对熵是最基本的概念，另两个可以从中推导出。与信息理论的大多数表述方式一样，我们从比较简单的熵概念开始。

B.1 熵

一个概率分布$P=(p_1,\cdots,p_n)$的熵$\mathcal{H}(P)$的定义如下：

$$\mathcal{H}(P)=\mathbf{E}(-\log p)=-\sum_{i=1}^{n}p_i\log p_i \tag{B.1}$$

熵的单位取决于对数的底。当底为2时，熵的单位是比特。熵度量了由P描述的随机试验结果的先验不确定性或观察到输出时所获得的信息量。它也是在无噪声条件下传输结果所需要的最小平均比特数（对数底取2）。

熵的概念可以从公理导出。考虑一个随机变量X，假设它以概率$p_1,\cdots,p_n$取值$x_1,\cdots,x_n$。目标是定义一个量$\mathcal{H}(P)=\mathcal{H}(X)=\mathcal{H}(p_1,\cdots,p_n)$，它能够以惟一方式度量概率分布所代表的不确定程度。值得注意的是：在给定与标尺对应的常数因子后，只需要三条常识性的公理，就足以惟一确定$\mathcal{H}$。这三条公理实际上组成一个整体规则。这三条公理如下：

1. $\mathcal{H}$是p_i的连续函数。

2. 如果所有的p_i相等，那么$\mathcal{H}(P)=\mathcal{H}(n)=\mathcal{H}(1/n,\cdots,1/n)$是$n$的单调递增函数。
3. 结合律：把所有的事件x_i分成k个不相交的类。A_i表示第i类中的所有事件的编号，$q_i=\sum_{j\in A_i}p_j$表示第i类对应的概率，则

$$\mathcal{H}(P)=\mathcal{H}(Q)+\sum_{i=1}^{k}q_i\,\mathcal{H}\left(\frac{\overline{P}_i}{q_i}\right) \tag{B.2}$$

其中的$\overline{P}_i$表示概率p_j的集合，$j\in A_i$。例如将以下的前两个事件合并为一组，结合律要求

$$\mathcal{H}(1/3,1/6,1/2)=\mathcal{H}(1/2,1/2)+\frac{1}{2}\mathcal{H}(2/3,1/3) \tag{B.3}$$

根据第一个条件，只要能对$p_i=n_i/n$, $i=1,\cdots,n$中的有理数情形确定$\mathcal{H}$就够了。[13] 根据第二和第三个条件，有

$$\mathcal{H}\left(\sum_{i=1}^{n}n_i\right)=\mathcal{H}(p_1,\cdots,p_n)+\sum_{i=1}^{n}p_i\mathcal{H}(n_i) \tag{B.4}$$

例如：

$$\mathcal{H}(9)=\mathcal{H}(3/9,4/9,2/9)+\frac{3}{9}\mathcal{H}(3)+\frac{4}{9}\mathcal{H}(4)+\frac{2}{9}\mathcal{H}(2) \tag{B.5}$$

特别地，设所有的n_i等于m，根据（B.4）我们得到

$$\mathcal{H}(m)+\mathcal{H}(n)=\mathcal{H}(mn) \tag{B.6}$$

由这个条件可以确定惟一解

$$\mathcal{H}(n)=C\ln n \tag{B.7}$$

其中$C>0$。代入（B.4），我们最终得到

$$\mathcal{H}(P)=-C\sum_{i=1}^{n}p_i\log p_i \tag{B.8}$$

对数的底由常量C确定。对数的底取2，则熵和信息量的单位是比特。在大多数情况下，我们在计算式中取自然对数，使得$C=1$。

[13] $p_i=n_i/n$似乎应为$p_i=n_i/\sum_i n_i$。——译者注

不难验证熵具有如下性质：

- $\mathcal{H}(P) \geqslant 0$
- $\mathcal{H}(P|Q) \leqslant \mathcal{H}(P)$，当且仅当$P$和$Q$独立时等号成立。
- $\mathcal{H}(P_1,\cdots,P_n) \leqslant \sum_{i=1}^{n}\mathcal{H}(P_i)$，当且仅当$P$和$Q$独立时等号成立。
- $\mathcal{H}(P)$ 在P上是凸（∩）的。
- $\mathcal{H}(P_1,\cdots,P_n) = \sum_{i=1}^{n}\mathcal{H}(P_i|P_{i-1},\cdots,P_1)$
- $\mathcal{H}(P) \leqslant \mathcal{H}(n)$，当且仅当$P$是均匀分布时等号成立。

B.2　相对熵

两个分布$P=(p_1,\cdots,p_n)$和$Q=(q_1,\cdots,q_n)$，或者对应的随机变量X和Y之间的相对熵定义为

$$\mathcal{H}(P,Q)=\mathcal{H}(X,Y)=\sum_{i=1}^{n}p_i\log\frac{p_i}{q_i} \tag{B.9}$$

相对熵也称为交叉熵，或者Kullback-Liebler距离或判别（相对熵的公理化表述可参看［486］及其中的参考文献）。它被看做P和Q之间距离的一种度量。P和Q之间越不相似，相对熵就越大。相对熵也可以测量所给出的信息量，它描述了一个假说相对于另一个假说的真实性。它也是对数似然度比值的期望值。严格地说，相对熵是不对称的，因此也不是一个距离。它可以通过取$\mathcal{H}(P,Q)+\mathcal{H}(Q,P)$来对称化，但在大多数情况下，不需要这种对称化的定义。如果用$U=(1/n,\cdots,1/n)$表示均匀分布，那么$\mathcal{H}(P,U)=\log n-\mathcal{H}(P)$。在这种意义上，熵是交叉熵的特殊情形。

用Jensen不等式（参见B.4节），可以证明的相对熵的以下两个重要性质：

- $\mathcal{H}(P,Q) \geqslant 0$，当且仅当$P=Q$时等号成立。
- $\mathcal{H}(P,Q)$ 在P和Q上是凸（∩）的。

第3章和第4章中关于统计力学和EM算法的部分使用了这些性质。

B.3　互信息

度量信息的第三个概念是互信息，考虑两个分布P和Q，以及它们在积空间上的联合分布R。互信息$\mathcal{I}(P,Q)$是联合分布R与边缘分布P和Q的积之间的相对熵：

$$\mathcal{I}(P,Q)=\mathcal{H}(R,PQ) \tag{B.10}$$

从上式可知它总是正值。当R是可分解的时候，即等于边缘分布的积时，互信息为0。互信息是相对熵的特殊情形。类似地，熵［或自熵（self-entropy）］是互信息的特殊情形，因为$\mathcal{H}(P)=\mathcal{I}(P,P)$。更进一步，互信息满足下列性质：

- $\mathcal{I}(P,Q)=0$，当且仅当P和Q相互独立。
- $\mathcal{I}(P_1,\cdots,P_n,Q)=\sum_{i=1}^{n}\mathcal{I}\left(P_i,Q|P_1,\cdots,P_{i-1}\right)$

用贝叶斯理论容易理解互信息：它代表观察到另一个变量时，某个变量的不确定性的减少量，也就是先验分布和后验分布的不确定性的差值。如果我们把两个变量分别记为X和Y，X的不确定性由它的熵来度量，即$\mathcal{H}(X)=\sum_x \mathbf{P}(X=x)\log\mathbf{P}(X=x)$。一旦我们观察到$Y=y$，则$X$的不确定性就是其后验分布的熵，即$\mathcal{H}(X|Y=y)=\sum_x \mathbf{P}(X=x|Y=y)\log\mathbf{P}(X=x|Y=y)$。这是一个依赖于观察值$y$的随机变量。它对所有可能的$y$的平均值称为条件熵：

$$\mathcal{H}(X|Y)=\sum_y P(y)\,\mathcal{H}(X|Y=y) \tag{B.11}$$

因此熵和条件熵之间的差值衡量了由Y的观察值提供的关于X的平均信息量。容易验证

$$\begin{aligned}\mathcal{I}(X,Y)=&\mathcal{H}(X)-\mathcal{H}(X|Y)=\\&\mathcal{H}(Y)-\mathcal{H}(Y|X)=\mathcal{H}(X)+\mathcal{H}(Y)-\mathcal{H}(Z)=\mathcal{I}(Y,X)\end{aligned} \tag{B.12}$$

其中$\mathcal{H}(Z)$是联合变量$Z=(X,Y)$的熵。也可以用对应的分布表示：

$$\begin{aligned}\mathcal{I}(P,Q)=&\mathcal{H}(P)-\mathcal{H}(P|Q)=\\&\mathcal{H}(Q)-\mathcal{H}(Q|P)=\mathcal{H}(P)+\mathcal{H}(Q)-\mathcal{H}(R)=\mathcal{I}(Q,P)\end{aligned} \tag{B.13}$$

可以画出与这些关系对应的经典Venn图，这作为练习留给读者。

B.4 Jensen不等式

本书多次用到Jensen不等式。如果一个函数f是凸（∩）的，并且X是一个随机变量，则有

$$\mathbf{E}f(X)\leqslant f\,\mathbf{E}(X) \tag{B.14}$$

进一步地，如果f是严格凸的，则等号成立就蕴涵着X是常量。如果用重心的概念来理解，这个不等式在图形上就显而易见了。$f(x_1),\cdots,f(x_n)$的重心低于$f(x^*)$，其中x^*是$x_1,\cdots,x_n$的重心。作为一种重要的特殊情况，$\mathbf{E}\log X\leqslant\log\mathbf{E}(X)$。

由此可以马上得到相对熵的性质。

B.5 最大熵

第2章和第3章讨论过在离散分布情况下的最大熵准则。连续情况下最大熵准则的精确表述更复杂一些。[282] 但是在任何情况下，如果我们把具有分布密度P的随机变量X的微分熵（differential entropy）定义为

$$\mathcal{H}(X) = -\int_{-\infty}^{+\infty} p(x)\log P(x)\, dx \tag{B.15}$$

则在所有具有方差σ^2的密度函数中，高斯分布$\mathcal{N}$（μ, σ）具有最大的微分熵。任意平均值和方差σ^2的高斯分布的微分熵是［$\log 2\pi e\sigma^2$］/2。考虑n维空间中的一个随机变量X，其向量均值为μ，方差矩阵为C，密度为P，则P的微分熵满足

$$\mathcal{H}(p) \leqslant \frac{1}{2}\log(2\pi e)^n |C| = \mathcal{H}\big(\mathcal{N}(\mu, C)\big) \tag{B.16}$$

当且仅当X的分布几乎处处符合$\mathcal{N}$（μ, C）时等号成立。此处的$|C|$表示C的行列式。

使用统计力学中波耳兹曼-吉布斯分布的推导，可以简单证明上述结果。例如，在一维情况下，高斯分布可以看做能量为$\mathcal{E}(x) = (x-\mu)^2/2\sigma^2$、分割函数为$\sqrt{2}\pi\sigma$、温度为1的波耳兹曼-吉布斯分布。因此给定惟一约束为能量期望的观察值后，高斯分布必然具有最大的熵。平均能量由$\int (x-\mu)^2 / 2\sigma^2 P(x)dx$给出，它是一个常量，一个等价的描述是标准方差为常量且等于σ。

这个结果可以推广到指数分布族中。在Dirichlet分布的情况下，考虑所有n维分布$P=(p_1,\cdots,p_n)$的空间。假设给定一个固定分布$R=(r_1,\cdots,r_n)$，用分布P与R之间的距离（相对熵）定义分布P的能量：

$$\mathcal{E}(P) = \mathcal{H}(R, P) = \sum_i r_i \log r_i - \sum_i r_i \log p_i \tag{B.17}$$

如果我们观察到的都是$\mathcal{E}$的平均值D，则对应的P的最大熵分布是波耳兹曼-吉布斯分布

$$\mathbf{P}(P) = \frac{e^{-\lambda\mathcal{E}}}{Z} = \frac{e^{-\lambda\,\mathcal{H}(R,P)}}{Z} = \frac{e^{\lambda\,\mathcal{H}(R)}\prod_i p_i^{\lambda r_i}}{Z(\lambda, R)} \tag{B.18}$$

其中λ是温度，依赖于平均能量的值D。现在，如果我们令$\alpha = \lambda+n$，$q_i = (\lambda r_i+1)/(\lambda+n)$，这个分布实际上就是参数为$\alpha$和$Q$的Dirichlet分布$\mathcal{D}_{\alpha Q}(P)$（注意到$\alpha \geqslant 0$，$q_i \geqslant 0$且$\sum_i q_i = 1$）。若$r_i$是均匀分布的，则$q_i$也是均匀分布。所以，所有的Dirichlet分布都可以看做最大熵的计算结果。

B.6 最小相对熵

最小相对熵准则[486]表示：若给定一个先验分布Q，应该选择一个分布P，使其满足问题的所有约束，并且使得相对熵$\mathcal{H}(P, Q)$最小化。而最大熵准则显然可以看做Q为均匀分布情况下的最小相对熵准则的特例。前面已提到，最小相对熵是用于寻找后验分布的准则，或者在先验分布中选择一个特殊的类。只有贝叶斯理论才能够正确估计后验分布，因此最小相对熵准则（或最大熵准则）不可能总是普适的。事实上，在一些例子中，最大熵似乎给出了“错误”的结果。[229]因此我们认为，不太可能存在一个确定先验分布的普适原则。如果确实需要这样的一个原则，它应该是：任何模型的最基本的先验分布都应该是均匀分布。换言之，在任何建模工作中，总是隐含地存在多级先验假设，最低一级的先验假设应该总为典型的均匀分布。在一些例子中，最小相对熵准则和贝叶斯的MAP估计能得到相同结果，建议仔细阅读这些很有启发意义的例子（参看第3章）。

附录C　概率图模型

C.1　符号和预备知识

在此附录中，主要复习一下概率图模型的基本理论[557,348]及其相应的高维概率分布的因子分解。首先介绍符号。如果X和Y是两个独立的随机变量，则记为$X \perp Y$。关于随机变量Z的条件独立，我们定义为$X \perp Y | Z$，即为$\mathbf{P}(X, Y|Z) = \mathbf{P}(X|Z)\mathbf{P}(Y|Z)$。特别要注意，随机变量的条件独立并不意味着边缘独立，反之也不成立。顶点集为V、边集为E的图记为$G=(V, E)$。顶点集编号为$V=\{1,2,\cdots,n\}$。如果是有向图，我们记为$G=(V, \vec{E})$。在考虑的所有图中，任意两个顶点之间最多只有一条边，顶点和它自己之间不存在边。在一个无向图中，$N(i)$表示所有与顶点i相邻的顶点组成的集合，而$C(i)$表示所有与顶点i存在通路的顶点所组成的集合。于是有

$$N(i)=\{j \in V: (i,j) \in E\} \tag{C.1}$$

如果一个图的任意顶点对之间存在一条边，则称此图为完全图。一个图G的团（clique）是G的最大完全子图。图G的团图（clique graph）G^C，其顶点对应于图G的每个团，两个顶点之间有边相连，当且仅当相应的团有非空的交集。

在一个有向图中，边的方向通常代表因果关系或时间不可逆性。我们用符号$N^-(i)$和$N^+(i)$分别定义顶点i的所有父节点和所有子节点。同样地，$C^-(i)$和$C^+(i)$分别定义顶点i的祖先（或“过去”）和后代（或“将来”）。所有这些概念显然可以推广到任意的顶点集合I中。对任意的$I \in V$，

$$N(I)=\{j\in V: i\in I \quad 且 \quad (i,j)\in E\}-I \tag{C.2}$$

上式也称为I的边界。在一个无向图中，当且仅当集合I与J不相交，并且I中的任意顶点到J中的任意顶点的任意路径包含K中的一个顶点时，顶点集I与顶点集J被顶点集K分离。

我们感兴趣的是形式为$\mathbf{P}(X_1,\cdots,X_n)$的高维概率分布，其中变量$X$表示隐变量或观察到的变量。特别地，我们最为感兴趣的是如何将这种高维分布因子分解为简单分布的乘积，如条件分布和边缘分布。显然，联合分布可以用边缘分布表示：

$$\mathbf{P}(X_1,\ldots,X_n)=\prod_{i=0}^{n-1}\mathbf{P}(X_{i+1}|X_1,\ldots X_i) \tag{C.3}$$

如果完全条件分布$\mathbf{P}(X_i|X_j: j\neq i)$集合是一致的，则它也惟一定义了联合分布（否则联合分布不能由条件分布确定）。[68,20]边缘分布$\mathbf{P}(X_i)$的完全集通常很不足以定义联合分布，除非有特殊情况（参见下文的因子分布）。如何由条件分布和边缘分布的一个任意集合惟一确定一个多变量联合分布，这个问题在参考文献［198］中讨论。我们会看到，图模型对应于联合分布，这种联合分布能方便由局部条件分布或一小类变量的联合分布表达。利用图模型概率推理可以逼近一些有用的分布，如后验分布等。许多技术被典型地应用于实现近似推理，这些技术包括概率传播、蒙特卡罗方法、统计力学、变分法和反向模型等。

由于技术上原因，[557]假设$\mathbf{P}(X_1,\cdots,X_n)$恒为正。由于可以赋予稀有事件很小的非零概率值，因此这个假设在实际应用中没有什么限制。考虑图$G=(V,E)$或$G=(V,\vec{E})$，其中变量X_i与相应的顶点i相关，定义X_I为变量$X_i: i\in I$的集合，I是指标集。对于一个固定的图G，$\mathcal{P}(G)$表示一组概率分布，各变量间的独立性由图中边的连续状态体现。简单地讲，缺少一条边意味着存在一个独立关系。对于有向图和无向图两种情形，这些独立关系在以下的两节中进行了精确定义。在建模中，对于任意的图G，实际概率分布可以不属于集合$\mathcal{P}(G)$。然而建模的目标是找到一个图G和$\mathcal{P}(G)$中的一个成员，使其尽可能地接近实际概率分布，其中逼近程度可以用相对熵等进行度量。

C.2 无向情形：马尔可夫随机域

在无向情形下，分布族$\mathcal{P}(G)$对应于马尔可夫随机域（Markov random field）、马尔可夫网络（Markov network）或概率性独立网络（probabilistic independence

network），有时还对应于波耳兹曼机（Boltzmann machine）等。[272,2] 对称相互作用模型在统计力学中应用较多，例如Ising模型和图像处理[199,392] 等，其中的联系更强调的是相关性而不是因果关系。

C.2.1 马尔可夫特性

图G的马尔可夫随机域可以用以下三种等价的马尔可夫独立性中的任意一种来刻画。三种马尔可夫独立性之间的等价性是明显的，它的证明留做练习。

1. **相邻马尔可夫特性**（pairwise Markov property）:不相邻的随机变量对X_i和X_j是否独立取决于其他所有随机变量，即对任意的$(i,j)\notin E$，有

$$X_i \perp X_j \big| X_{V-\{i,j\}} \tag{C.4}$$

2. **局部马尔可夫特性**（local Markov property）：给定相邻顶点时，任意随机变量X_i与其他所有的随机变量独立，即对V中任意的i，有

$$X_i \perp X_{V-N(i)\cup\{i\}} \big| X_{N(i)} \tag{C.5}$$

3. **全局马尔可夫特性**（global Markov property）:如果I和J是两个被K分离的不相交的顶点集，对应的随机变量集在给定第三个集合的随机变量时条件独立：

$$X_I \perp X_J \big| X_K \tag{C.6}$$

这些独立性等价于以下情形：

$$\mathbf{P}\left(X_i \big| X_{V-\{i\}}\right) = \mathbf{P}\left(X_i \big| X_{N(i)}\right) \tag{C.7}$$

C.2.2 因子分解性

函数$\mathbf{P}(X_i|X_j: j\in N(i))$称为马尔可夫随机域的局部特征。通过一种复杂的方法，这组函数可以惟一确定全局分布$\mathbf{P}(X_1,\cdots,X_n)$。特别地，与有向情形不同，全局分布并不是所有局部特性的乘积。然而有一个重要的定理将马尔可夫随机域与波耳兹曼-吉布斯分布相联系。作为局部独立性的一个结果，可以证明马尔可夫随机域的全局分布有以下的函数形式：

$$\mathbf{P}\left(X_1,\cdots,X_n\right) = \frac{e^{-f(x_1,\cdots,X_n)}}{Z} = \frac{e^{-\Sigma_C f_C(X_C)}}{Z} \tag{C.8}$$

其中Z是通常的归一化因子。C取遍图G的所有团。f_C称为团C的势函数（potential function）或团函数，它仅依赖于出现在相应团中的变量X_C。f称为能量。事实上，当且仅当（C.8）成立时，$\mathbf{P}$和G确定一个马尔可夫随机域。[500]

通过定义并结合波耳兹曼-吉布斯表示，容易从团的势函数中推导出局部特征和边缘分布。另一方面，势函数不是惟一的。虽然一般情况下，确定一组势函数是一个非常细致的过程，但仍有从局部特征推导势函数的公式。有一种重要的特殊情形特别简单，即当图是三角化（triangulated）时的情形。如果一个图G中长度大于等于4的圈包含至少1个旋（chord），则称此图是三角化的。简单连接图（即树）是三角化图的一个重要特例。一个图是三角化的，当且仅当它的团图满足所谓运行相交性（running intersection property）的特殊性质。此性质规定，如果图G的一个顶点属于G的两个团C_1和C_2，那么此顶点一定也属于团图G^C中所有在从C_1到C_2的路径上的其他团。图G的两个相邻团C_1和C_2（即团图G^C中的两个相邻节点）的交集，称为一个分离子（separator）。对于三角化图，这个分离于将C_1和C_2分离，使它们条件独立。

三角图的另外一个重要性质是完美编号（perfect numbering）。如果对于所有的i，$N(i)\cap\{1,2,\cdots,i-1\}$这些顶点构成的图是完全图，那么对于$V$中各节点的这种编号是完美的。一个图是三角化的，当且仅当此图存在一个完美编号（见参考文献［512，350］及其中的参考文献）。这里的关键点是针对与三角图有关的马尔可夫随机域，其全局分布有以下的形式：

$$\mathbf{P}(X_1,\cdots,X_n)=\frac{\prod_C \mathbf{P}(X_C)}{\prod_S \mathbf{P}(X_S)} \tag{C.9}$$

其中C和S分别取遍出现在交叉树中的团和分离子，所谓交叉树是G^C的最大生成树（maximal spanning tree）。$\prod_C \mathbf{P}(X_C)$是X_C的边缘联合分布。团的势函数因此是明显的。

当图G没有边时，马尔可夫随机域一种非常特殊的情形，即所有的变量独立且$\mathbf{P}(X_1,\cdots,X_n)=\prod_{i=1}^n \mathbf{P}(X_i)$。这样的联合分布或马尔可夫随机域称为析因的（factorial）。给定多元联合分布P，容易发现，从相对熵角度考察，在所有的析因分布中，与P最为接近的是P的边缘的乘积。

C.3 有向情形：贝叶斯网络

在有向情形下，分布族$\mathcal{P}(G)$对应于贝叶斯网络、置信网络、有向概率独立

网络、有向马尔可夫域、因果网络（causal network）、影响图表（influence diagram），甚至马尔可夫网络（Markov mesh）等概念（一个简单的分子生物学图示见参考文献［322］）。[416,557,121,106,286,246] 上面提及有向图的边的方向，通常表示因果关系或时间不可逆性。这样的模型是常见的，例如在设计专家系统中。

在有向情形下，我们有一个有向图$G=(V,\vec{E})$。这个有向图也被假定为无环的（acyclic），即没有有向环。这是因为从局部条件概率的乘积出发，不可能在环上一致地定义变量的联合概率。也就是说，在一般情况下，乘积$\mathbf{P}(X_2|X_1)\mathbf{P}(X_3|X_2)\mathbf{P}(X_1|X_3)$不能一致地确定$X_1$，$X_2$，$X_3$的分布。一个无环有向图表示了一种偏序关系。特别地，有可能对顶点进行编号，使得如果从i到j有一条边，则有偏序关系$i<j$。换言之，联系边的偏序关系与编号方式一致。这种序关系也称为一个拓扑类。我们将在任何必要时候选择采用这种序关系，使得$C^-(i)$的过去包含在$\{1,2,\cdots,i-1\}$中，而$C^+(i)$的将来包含在$\{i+1,\cdots,n\}$中。有向图$G=(V,\vec{E})$的伦图（moral）是一个无向图$G^M=(V,E+M)$，它通过以下方式得到：去除G中边的方向，并加入一些边，使图G中同一节点的任意两个父节点保持连通。伦图这个概念是参考文献［350］首先引入的，用以描述所有的“父辈”都已“结婚”。现在我们可以描述基本的有向无环图模型的马尔可夫独立性。

C.3.1 马尔可夫特性

一个有向无环图G的贝叶斯网络，可以通过一系列等价独立性中的任意一个加以刻画。在所有情况下，有向情形时的基本马尔可夫思想是：以现在为条件，将来独立于过去。或者说，为了预测将来，所有相关的信息只从现在获得。

相邻马尔可夫特性

满足$i<j$的不相邻的随机变量对X_i和X_j是否独立，取决于j的过去中所有别的随机变量。即对于任意的$(i,j)\notin\vec{E}$且$i<j$，有

$$X_i \perp X_j \Big| X_{C^-(j)-\{i\}} \tag{C.10}$$

事实上可以用更大的集$\{1,\cdots,j-1\}$代替$C^-(j)$。另一个等价的表述是：在一个节点集I中，X_i与X_j独立，当且仅当i和j是d分离的（d-separated），即不存在从i到j的d连通路径（d-connecting）。[121] 下面给出从i到j的d连通路径的定义。考虑从i到j的路径中一个节点k，对应路径中经过节点k的两条边的方向是一进一出、都是出或都是进，分别称节点k是线性、发散或收敛的。在节点集I中，从i到j的路径是d连通的，当且仅当此路径中每一个内部节点k或者是线性或发散并且不包含在I中，

或者是收敛并且满足［$k \cup C^+(k)$］$\cap I \neq \varnothing$。直观上看，i和j是d连通的，当且仅当或者在i和j之间有一条因果路径，或者在I中有证据（evidence）补偿相互关联的两个节点。

局部马尔可夫特性

除了它的后代外，在它的父节点中，一个随机变量X_i独立于所有其他节点的随机变量。因此若$j \notin C^+(i)$且$j \neq i$，有

$$X_i \perp X_j \big| X_{N^-(i)} \tag{C.11}$$

全局马尔可夫特性

给定I和J是两个不相交的顶点集合，在有向图G中称K分离I和J，当且仅当在包含I、J和K的最小祖先集的无向伦图（moral undirected graph）中，K分离I和J。[349]全局马尔可夫特性与这一个分离概念一样，即若K分离I和J，有

$$X_I \perp X_J | X_K \tag{C.12}$$

参考文献［557］中证明，有向图G也具有相应的伦图G^M所具有的所有马尔可夫独立关系。一般来说反之不成立，除非G^M是通过仅仅除去图G中边的方向而得到，也就是说没有在父节点间添加新的边。最后，三种马尔可夫独立性中的任何一种等价于以下公式：

$$\mathbf{P}\left(X_i \middle| X_{C^-(i)}\right) = \mathbf{P}\left(X_i \middle| X_{N^-(i)}\right) \tag{C.13}$$

事实上，可以用更大的集$\{1,\cdots,i-1\}$代替$C^-(i)$。

C.3.2 因子分解性

作为马尔可夫特性的一个推论，不难看出单向局部特征$\mathbf{P}(X_i|X_{N^-(i)})$相互一致。事实上，给定一个图，这些局部特征惟一确定关于此图的贝叶斯网络。实际上我们有

$$\mathbf{P}(X_1,\cdots,X_n) = \prod \mathbf{P}\left(X_i \middle| X_{N^-(i)}\right) \tag{C.14}$$

这是一个基本性质。局部条件概率可以通过查询表（lookup table）确定，尽管由于查询表太大，通过它确定局部条件概率通常不切实际。通常会采用一些更紧凑的但缺乏一般性的表示法，例如噪声OR网络[416]或神经网络表示法，而对于二元变量，则可以采用sigmoidal置信网络[395]，后者的特征由局部连通权重和

sigmoidal函数确定，还可以利用归一化指数函数推广到多变量的情形。混合模型参数化的另一例子是在每一个顶点有一个局部神经网络用于计算局部特征。

C.3.3 学习和传播

在一般的图模型和特殊的贝叶斯网络中，从学习图结构本身到通过数据学习局部条件分布，有几个学习的层次水平。除C.3.6节外，这里不再讨论此问题，文献的综述及参考可以在参考文献［106, 246］中找到。另一个与贝叶斯网络有关的基本操作是证据传播（propagation of evidence），即给定观察到的节点变量时，更新每一个变量X_i的条件概率。证据传播在一般的情形下是NP完全的（NP-complete）。[135] 但利用一种简单的消息传递方法，[416,4] 简单连通图（即无向图中任意两节点之间的路径数目不超过1）的传播时间是节点数n的线性函数。对于一般情形，可以利用原图的三角化伦图中的闭，对原始变量进行聚类，从而构造与原有网络等价的简单连通图——交叉树。已有的用多连通网络的精确算法都依赖于这种简化。（参考文献［416, 350, 467］中给出了算法，参考文献［287］对此做了进一步改进。）

参考文献［145］给出了一种类似的算法，用于估计变量X_i最有可能的概率分布结构。沙克特（Schachter）等人在参考文献［468］中证明了：在某种意义上，所有精确的推理算法都与参考文献［287］和［145］中的算法等价。一个重要的猜想是，参考文献［416］中的简单的消息传递方法对于多连通网络也能得到很好的近似。这个猜测得到编码理论的实验观察和推导理论的支持（详见参考文献［385］）。

C.3.4 一般特性

值得注意的是，本书中所用的大多数模型都可以看做贝叶斯网络的实例。前馈人工神经网络是一种贝叶斯网络，其中局部条件概率函数是德尔塔函数。同样地，HMM和马尔可夫系统一般有非常简单的贝叶斯网络表示方式。事实上，HMM是马尔可夫随机域和贝叶斯网络的一种特例。我们把推导贝叶斯网络表示方式的过程留给读者作为练习，许多诸如混合分布、分层先验估计、Kalman滤波器和其他状态空间模型概念等的贝叶斯网络表示方法也留给读者作为练习。贝叶斯网络表示方式的一般性是许多正在研究的模型类的基础。这也适合于HMM的许多推广形式，例如输入—输出HMM（见第9章）、树结构HMM[293] 和因子HMM。[205]

将一般的贝叶斯网络传播算法应用到特殊情形，人们重新推导出一些常用的算法。例如对于HMM，人们从Pearl算法得到了通常的前向—后向算法和Viterbi

算法。[493]其他领域的研究同样如此，例如编码理论（快速编码和Gallager-Tanner-Wiberg解码）和Kalman滤波器理论（Rauch-Tung-Streibel光滑器）的研究，甚至确定性的组合算法（快速傅里叶变换）。[4,204]虽然并未仔细考察，但我们猜测上下文无关语法中的内部—外部算法也是一种特例。虽然证据传播算法通常是NP完全的，[210, 578]但是利用蒙特卡罗方法（如吉布斯采样）或变分法（如中值域理论，参见附录A和参考文献［465,276,204］），有时甚至限定特定问题的特殊网络结构，可以得到一些近似算法。吉布斯采样具有简单性和普遍性，它对贝叶斯网络显得特别具有吸引力。

C.3.5 吉布斯采样

观察到与可见节点关联的变量后，对于其他任意节点i，我们利用其他所有变量给定时的条件概率对其取值采样。根据因子分解公式（C.14），我们有

$$\mathbf{P}\left(X_i \middle| X_{V-\{i\}}\right)=\frac{\mathbf{P}(X_V)}{\mathbf{P}\left(X_{V-\{i\}}\right)}=\frac{\prod_j \mathbf{P}\left(X_j \middle| X_{N^-(j)}\right)}{\sum_{x_i}\mathbf{P}(X_1,\cdots,X_i=x_i,\cdots,X_n)} \tag{C.15}$$

通过化简分子和分母，由上式得到

$$\mathbf{P}\left(X_i \middle| X_{V-\{i\}}\right)=\frac{\mathbf{P}\left(X_i \middle| X_{N^-(j)}\right)\prod_{j\in N^+(i)}\mathbf{P}\left(X_j \middle| X_{N^-(j)}\right)}{\sum_{x_i}\mathbf{P}\left(X_i=x_i \middle| N^-(i)\right)\prod_{j\in N^+(i)}\mathbf{P}\left(X_j \middle| X_{N^-(j)}\right)} \tag{C.16}$$

正如期望那样，吉布斯采样所需的条件分布是局部的，它只依赖于节点i及其父节点和子节点。从而可通过在每个节点上对记数值进行平均得到后验估计，这样就只占用非常少的内存。可以通过在每个节点上对概率进行平均得到更加精确的估计（参考文献［396］对此有部分讨论）。进行吉布斯采样时，关键在于过程的周期（对于多次使用相同的采样器，则考察其重复时间）和放弃初始样本（“burn-in”），因为这些都无法代表平衡分布。

C.3.6 休眠—响应算法和Helmholtz机

在参考文献［255,146］中叙述了一类特殊的贝叶斯网络的条件分布学习算法，这个算法在理论上有意思但未必实用。这类贝叶斯网络由识别网络和生成网络这两个逆模型组成。始于输入层，识别网络有一个前馈层构造。所有隐层中的节点对应随机二值变量，更一般的情形也类似，例如对于多值单元。利用组合权重和Sigmoidal激活函数，可以在神经网络类型中实现局部条件分布。单元i的激

活概率由下式给出

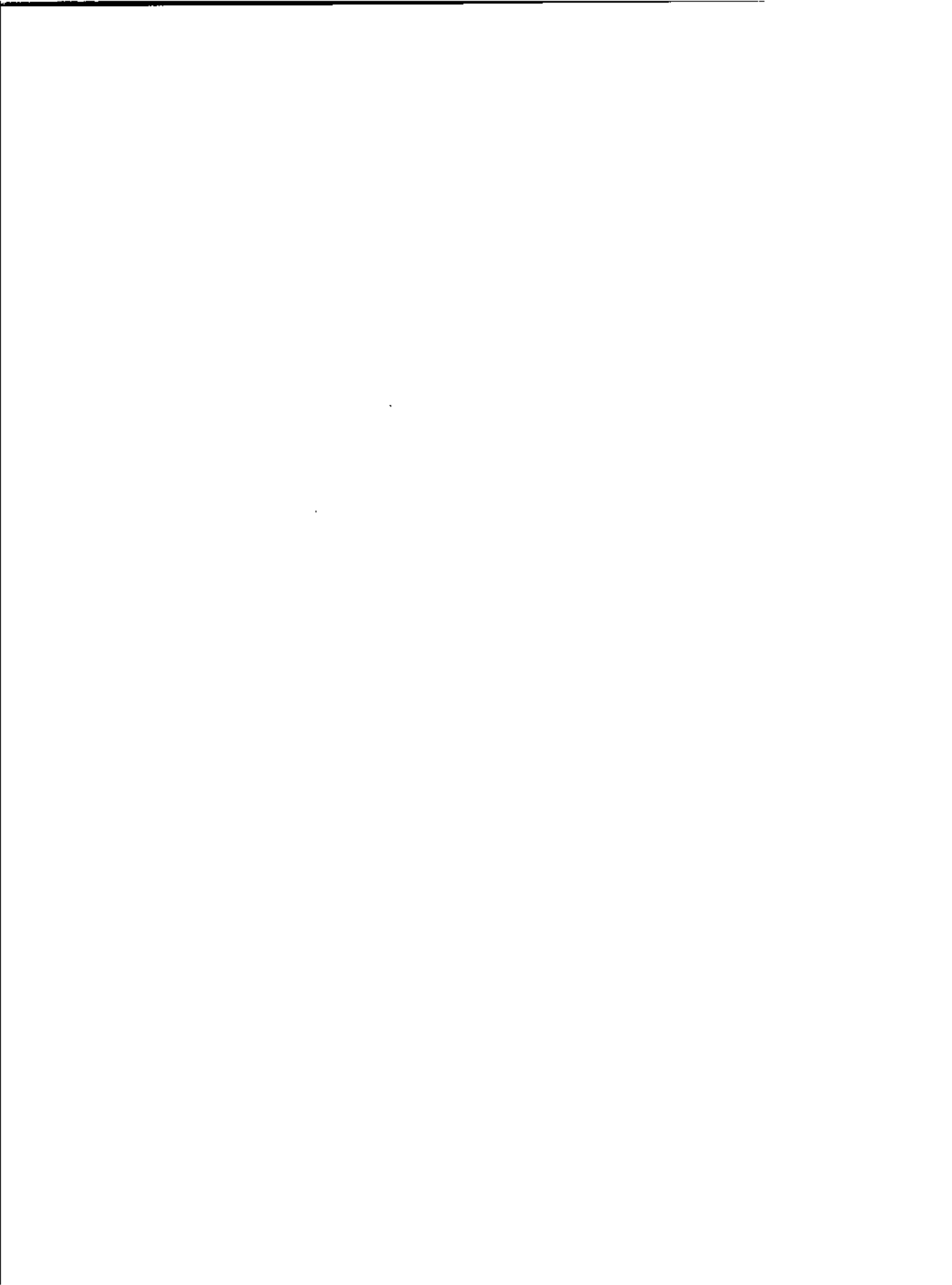
$$\mathbf{P}(X_i = 1) = \frac{1}{1 + e^{-\sum_{k \in N^-(i)} w_{ik} x_k + b_i^l}} \tag{C.17}$$

其中x_k为前一层节点的状态。生成网络则是识别网络的镜像。它是一个前馈分层网络，起始于识别网络的顶端隐层，终止于识别网络的输入层。它使用的节点也与识别网络相同，但带有一个逆向连接集。这些逆向连接集引入了局部环，结果在构造的联合网络中引入环。然而，这些环并不会对结果造成什么不良影响，因为学习过程中是交替而不是同时使用这两个网络。

根据公认的生物学解释命名的休眠—响应（sleep-wake）算法是前向和后向连接权重的一种非监督学习算法。此算法在两种状态之间进行切换。在每一种状态下，一个网络节点的输出成为另一网络的局部目标，通过德尔塔规则训练其权重。在响应状态（wake phase）下，识别网络被激活，并根据下式更新生成权重

$$\Delta w_{jk} = \eta x_k (x_j - p_j) \tag{C.18}$$

其中x_j代表识别网络中节点 j 的状态，p_j是根据生成连接由（C.17）确定的相应概率。休眠状态（sleep phase）则使用一种对称的更新规则，其中通过生成网络的输出用于修改识别网络的权重。[255,574]



附录D　HMM相关技术：标定、周期构架、状态函数和Dirichlet混合模型

D.1　标　定

我们已经指出，概率$\mathbf{P}(\pi|O,w)$的数值一般非常小，很容易超出机器的精度范围，对于前向变量$\alpha_i(t)$，随着t的增大，也会出现超出机器精度范围的情况。类似的情况也会发生在后向变量$\beta_i(t)$随t减小时。解决这个问题的方法是用一个仅依赖于t的适当的系数，标定时刻t的前向和后向变量。可以以互补的方式定义α和β的标定系数，使得训练公式经过标定后保持基本不变。我们下面将依照参考文献［439］的思路，给出前向变量和后向变量的精确标定公式。[14]为了简单起见，附录中考察的HMM只包含生成状态。我们把包含删除状态的一般公式的推导留给读者作为练习。

D.1.1　前向变量的标定

更准确地讲，我们定义一个新的标定变量

$$\hat{\alpha}_i(t)=\frac{\alpha_i(t)}{\sum_j \alpha_j(t)} \tag{D.1}$$

在时刻0，对于任意状态i,有$\alpha_i(0)=\hat{\alpha}_i(0)$。标定变量可以通过递归计算获得，

---

⑭ 参考文献［439］中的标定公式有些错误，作者附上了更正说明。

其中传播步骤和标定步骤交替进行。令 $\hat{\hat{\alpha}}_i(t)$ 表示对应于 $\hat{\alpha}_i(t)$ 的经过传播步骤但尚未经过标定的数值。假设所有变量均已计算至时刻t-1，我们首先利用（7.5）进行传播计算得到

$$\hat{\hat{\alpha}}_i(t)=\sum_{j\in N^-(i)}\hat{\alpha}_j(t-1)t_{ij}e_{iX^t} \tag{D.2}$$

其中 $\hat{\hat{\alpha}}_i(0)=\alpha_i(0)$。这与 $\alpha_i(t)$的传播公式形式相同。进一步利用（D.1）得到

$$\hat{\hat{\alpha}}_i(t)=\frac{\alpha_i(t)}{\sum_j\alpha_j(t-1)} \tag{D.3}$$

然后再对 $\hat{\hat{\alpha}}(t)$进行标定，我们发现对（D.3）标定等价于对α进行标定：

$$\frac{\hat{\hat{\alpha}}(t)}{\sum_j\hat{\hat{\alpha}}_j(t)}=\frac{\alpha_i(t)}{\sum_j\alpha_j(t)}=\hat{\alpha}_i(t) \tag{D.4}$$

上述公式中需要计算每一个迭代步骤的标定系数$c(t)=\sum_i\hat{\hat{\alpha}}_i(t)$。由（D.3），$c(t)$和$\alpha$的标定系数$C(t)=\sum_i\alpha_i(t)$之间的关系为

$$C(t)=\prod_{\tau=1}^{t}c(\tau) \tag{D.5}$$

D.1.2 后向变量的标定

后向变量的标定稍有不同，其中标定系数由前向传播计算获得，而不是通过β直接获得。尤其是这意味着，在后向传播开始之前必须首先计算前向传播。相应地，我们定义一个标定变量

$$\hat{\beta}_i(t)=\frac{\beta_i(t)}{D(t)} \tag{D.6}$$

其中标定系数定义为

$$D(t)=\prod_{\tau=t}^{T}c(\tau) \tag{D.7}$$

在下面我们会看到选择这一定义的原因。假设所有变量均已反向计算至（t+1）时刻，首先应用（7.10）对 $\hat{\beta}$ 进行反向传播计算得到

$$\hat{\hat{\beta}}_i(t)=\sum_{j\in N^+(i)}\hat{\beta}_j(t+1)t_{ji}e_{jX^{t+1}} \tag{D.8}$$

然后用$c(t)$对$\hat{\hat{\beta}}_i(t)$进行标定，得到

$$\hat{\beta}_i(t)=\frac{\hat{\hat{\beta}}_i(t)}{c(t)}=\frac{\beta_i(t)}{D(t)} \tag{D.9}$$

这正如（D.6）所要求的形式。

D.1.3 学习过程

下面考虑任意一种学习算法的公式，如计算转移参数的EM学习算法公式（7.31）：

$$t_{ji}^{+}=\frac{\sum_{t=0}^{T}\gamma_{ji}(t)}{\sum_{t=0}^{T}\gamma_i(t)}=\frac{\sum_{t=0}^{T}\alpha_i(t)t_{ji}e_{jX^{t+1}}\beta_j(t+1)}{\sum_{t=0}^{T}\sum_{j\in S}\alpha_i(t)t_{ji}e_{jX^{t+1}}\beta_j(t+1)} \tag{D.10}$$

任何形如$\alpha_i(t)\beta_j(t+1)$的乘积等价于$C\hat{\alpha}_i(t)\hat{\beta}_j(t+1)$，其中$C=C(t)\ D(t+1)=\prod_1^T c(t)$并且与$t$无关。而分子和分母中的常数$C$都被约去了。因此只需用相应的标定变量$\hat{\alpha}$和$\hat{\beta}$替代原公式中的$\alpha$和$\beta$就可以直接使用相同的学习算法公式。类似的推导过程可应用于其他学习算法。

D.2 周期构架

D.2.1 轮状构架

在第8章讨论的轮状构架（wheel architecture）中，我们可以设想有一个初始状态与轮上所有状态相连接。类似地，可以设想也有一个终止状态与轮上所有状态相连接。轮状构架不包含删除状态，因此所有算法（前向、后向、Viterbi以及标定）都将被简化，即不再需要区分生成状态和删除状态。

D.2.2 环状构架

环状构架（loop architecture）比轮状构架更具一般性，因为它包含删除状态，甚至包含经过删除状态的循环路径的概率。我们引入如下表示法：

- h表示环的锚状态。锚状态是一个删除（哑）状态，尽管它不与任何主状态相关联。
- L表示环中的状态集合。
- κ表示沿环移动一周期而不生成任何符号的概率。它是环中所有与连续的删除状态相关的t_{ji}的乘积。

- t_{ji}^d表示构架中从状态i到状态j的最短哑路径的概率。
- t_{ji}^D表示从状态i移动到状态j而不生成任何符号的概率。对于任意两个状态，若连接它们的路径中至少有一条包含锚状态，则有 $t_{ji}^D = t_{ji}^d\left(1+\kappa+\left(\kappa^2\right)\cdots\right) = t_{ji}^d/(1-\kappa)$。

前向传播公式

无论对于即时传播还是在平衡状态，前向传播公式都成立。对于任何生成状态$i \in E$，

$$\alpha_i(t+1) = \sum_{j \in N^-(i)} \alpha_j(t) t_{ij} e_{iX^{t+1}} \tag{D.11}$$

对于任意哑状态i以及锚状态

$$\alpha_i(t+1) = \sum_{j \in N^-(i)} \alpha_j(t+1) t_{ij} \tag{D.12}$$

对于锚状态，我们可以将来自环和来自旁侧部分的贡献分离：

$$\alpha_h(t+1) = \sum_{j \in N^-(h)-L} \alpha_j(t+1) t_{hj} + \sum_{j \in N^-(h) \cap L} \alpha_j(t+1) \tag{D.13}$$

实　现

实现上述传播算法有三种方式：第一，迭代即时传播公式直到平衡为止。第二，对于锚状态，只对平衡公式沿环状构架迭代一次。具体地讲，令$x=\alpha_h$（$t+1$），将该等式作为x的函数沿环状构架前向传播一次，最终求解x。当循环完成时，将获得一个形如$x=ax+b$的等式，于是$x=b/(1-a)$。然后，用这个新值代替x，代人表达式α_i（$t+1$），其中$i \in L$。第三，求x的解析解，即直接求$x=\alpha_h$（$t+1$）的平衡值（例如求上述a和b）。注意到产生表达式α_h（$t+1$）的路径，可以根据X^{t+1}生于环内部还是环外部分为两类：

$$\alpha_h(t+1) = \sum_{j \in N^-(h)-L} \alpha_j(t+1) t_{hj}\left(1+\kappa+\kappa^2+\cdots\right) + \sum_{j \in N^-(h) \cap L} \alpha_j(t+1) t_{hj}^D \tag{D.14}$$

等号右侧的第二项对应于在环内部生成X^{t+1}的情况，它包含任意数量的结束于锚状态的哑转移。这项中包含的未知变量α_j（$t+1$）可以由传播算法中上次迭代的结果α_j（t）计算获得。由此，我们最终可以得到

$$\alpha_h(t+1) = \frac{1}{1-\kappa} \sum_{j \in N^-(h)-L} \alpha_j(t+1) t_{hj} + \sum_{j \in E \cap L} \sum_{k \in N^-(i)} \alpha_k(t) a_{jk} e_{jX^{t+1}} t_{hj}^D \tag{D.15}$$

对于上式最后一个求和项的计算，我们使用如下方法——前向传播两个变量$\alpha_i(t)$和$\alpha_i^L(t)$。$\alpha_i^L(t)$可以被解释为状态i在时刻t的概率，这时环中已经产生了符号t，但尚未再次经历锚状态。对于环中的任意生成状态i，传播公式为

$$\alpha_i(t+1)=\alpha_i^L(t+1)=\sum_{j\in N^-(i)}\alpha_j(t)t_{ij}e_{iX^{t+1}} \quad (D.16)$$

对于环中的任意哑状态i（删除状态或锚状态），传播公式为

$$\alpha_i^L(t+1)=\sum_{j\in N^-(i)\cap L}\alpha_j^L(t+1)t_{ij} \quad (D.17)$$

这些公式应用$\alpha_h^L(t+1)=0$初始化，并沿所有路径通过环传播一次，最终得到$\alpha_h^L(t+1)$的新值。于是我们得到

$$\alpha_h(t+1)=\frac{1}{1-\kappa}\left[\sum_{j\in N^-(h)-L}\alpha_j(t+1)t_{hj}+\alpha_h^L(t+1)\right] \quad (D.18)$$

在时刻0，做如下初始化：

- $\alpha_i(0)=0$，对任意生成状态；
- $\alpha_i^L(0)=0$，对任意状态，包括锚状态；
- $\alpha_h(0)=\sum_{j\in N^-(h)-L}\alpha_j(0)t_{hj}/(1-\kappa)$
- $\alpha_i(0)=\sum_{j\in N^-(i)}\alpha_j(0)t_{ij}$，对环中除锚状态外的任意哑状态。

通过在循环中同时传播$\alpha(t)$和$\alpha^L(t)$，所有变量均可按以下顺序在沿环的一次传播计算中获得。在步骤t，假设对于定位状态和全部生成状态，$\alpha_i(t)$已知。具体步骤如下：

- 令$\alpha_h^L(t+1)=0$；
- 沿环同时前向传播：对哑状态由（D.12）计算$\alpha_i(t)$，对生成状态由（D.16）计算$\alpha_i(t+1)=\alpha_i^L(t+1)$，对所有哑状态还需由（D.17）计算$\alpha_i^L(t+1)$；
- 由（D.18）计算$\alpha_h(t+1)$。

用相同的方法可以导出环状构架的后向传播和标定公式。

D.3 状态函数：可弯曲性

正如第7章和第8章中所讨论的，任何依赖于家族中氨基酸或核苷酸的局部组成的函数，如熵、疏水性或可弯曲性等，都可以用HMM进行研究。尤其是可以通过HMM主干概率计算这类函数的数学期望，以加强一些在家族中单个成员序

列上通常难以显现的模式。只对单个字符（如熵、疏水性）定义相应的函数或标度时，很容易计算其数学期望。而当函数依赖于相邻的二元或三元字符时，计算会变得复杂一些，经常出现的情形是DNA的二核苷酸或三核苷酸［可弯曲性、核小体定位、堆积能、螺旋扭曲（propeller twist）等］。围绕HMM主干的这些函数能够帮助我们确定相应家族的结构和功能特性。目前的HMM仿真程序可以提供50种以上的函数。下面我们将就可弯曲性讨论如何计算这类数学期望，由于依赖于三元字符，因此与仅依赖于单个字符的函数相比，计算会困难一些。

D.3.1 动 机

为了避免引入外部干扰（exogenous artifact），我们可以通过多重序列比对直接计算平均可弯曲性的分布图。尽管多重序列比对方法很有用，我们仍希望能够通过HMM直接计算可弯曲性的分布图，具体原因如下：

- HMM计算得更快，因为一旦训练获得HMM后计算即可进行，而无需将所有序列与模型进行比对。
- 在我们试验过的许多情况下，由HMM产生的分布图与多重序列比对的结果具有非常相似的特征。两种分布图的一致性，进一步证明HMM对于给定数据是一个很好的模型；而两种分布图相差较大的情况，则能够给我们一些启示。
- 在某些情况下，例如数据很少时，正则化得很好的HMM能够产生更好的可弯曲性分布图。

D.3.2 HMM可弯曲性分布图的定义

下面我们将使用标准线性HMM构架，当然也可以使用环状或轮状构架进行类似的计算。在HMM可弯曲性分布图的定义中，很自然地只考虑HMM的主状态$m_0,\cdots$，m_{N+1}，其中m_0为初始状态，m_{N+1}为终止状态（除非转移到插入或删除状态的概率很高，否则在我们的计算中将不包含它们）。序列$O=\left(X_O^1,\cdots,X_O^N\right)$在位置$i$（远离边界时）的可弯曲性$B(i,O)$，可以被定义为在长度为$W=2l+1$的窗口上的平均三元可弯曲性：

$$B(i,O)=\frac{1}{W}\sum_{j=i-l}^{i+l-2}b\left(X_O^j,\cdots X_O^{j+2}\right) \tag{D.19}$$

其中b(X, Y, Z）表示在特定标度下XYZ的三元可弯曲性（见文献［96］及其参考文献）。序列家族在位置i的可弯曲性B（i）自然被定义为所有可能的主干序列的

平均值：

$$B(i)=\sum_{O}B(i,O)\mathbf{P}(O) \qquad (\text{D.20})$$

然而，以上公式的计算效率并不高，因为可能的序列数量是N的指数函数。幸好我们能找到更有效的方式进行计算。

D.3.3 可弯曲性分布图的高效算法

由（D.20）我们发现

$$B(i)=\sum_{O}B(i,O)\prod_{k=1}^{N}e_{k\mathrm{X}_O^k}\prod_{k=0}^{N+1}t_{m_k m_{k+1}} \qquad (\text{D.21})$$

最后一项是所有HMM主干转移概率的乘积，它等于某一常数C。再将（D.19）代入（D.21）得到

$$B(i)=\frac{C}{W}\sum_{O}\sum_{j=i-l}^{i+l-2}b\left(\mathrm{X}_O^j,\cdots,\mathrm{X}_O^{j+2}\right)\prod_{k=1}^{N}e_{k\mathrm{X}_O^k} \qquad (\text{D.22})$$

交换求和顺序可得

$$B(i)=\frac{C}{W}\sum_{j=i-l}^{i+l-2}\sum_{O}b\left(\mathrm{X}_O^j,\cdots,\mathrm{X}_O^{j+2}\right)\prod_{k=1}^{N}e_{k\mathrm{X}_O^k} \qquad (\text{D.23})$$

为了对所有序列求和，我们可以根据出现在位置j, j+1和j+2的符号X，Y和Z将序列分为不同的组。经过化简最终得到

$$B(i)=\frac{C}{W}\sum_{j=i-l}^{i+l-2}\sum_{\mathrm{X,Y,Z}}b(\mathrm{X,Y,Z})e_{j\mathrm{X}}e_{j+1\mathrm{Y}}e_{j+2\mathrm{Z}} \qquad (\text{D.24})$$

因此，（D.20）中的定义等价于（D.24）中的定义，它是对窗口中所有可能的三元字符进行的求和运算，这个求和是以在相应位置的生成概率的乘积作为权重的。定义（D.24）显然是最容易实现的，我们已经利用这一定义从HMM计算过许多可弯曲性的分布图，其中常常忽略了恒定的标度因子C/W。一般而言，边界的影响仅限于最初和最终l个状态。

D.4 Dirichlet混合模型

首先回顾一下第2章和第3章的内容。Dirichlet分布$\mathcal{D}_{\alpha Q}(P)$的均值为$Q$，最大值为$p_\mathrm{X}$=（$\alpha q_\mathrm{X}-1$）/（$\alpha-|A|$），其中对于所有X有$p_\mathrm{X}\geqslant 0$。一个混合Dirichlet分布

定义为$\mathbf{P}(P)=\sum_1^n\lambda_i\,\mathcal{D}_{\alpha_iQ_i}(P)$，其中混合系数须满足$\lambda_i\geq 0$及$\sum_i\lambda_i=1$。根据数学期望的线性特性，混合分布的数学期望为$\sum_i\lambda_iQ_i$。对于一模型，我们一般无法一般性地确定最大值的解析表达式。

D.4.1 Dirichlet混合先验分布

考虑如何为与HMM生成状态（或等价的比对列上的骰子模型）相关的生成分布$P=(p_X)$选择一个先验分布。这里p_X为模型的参数。数据集D由在该列观察到字符出现的次数$D=(n_X)$构成，并有约束条件$\sum_X n_X=N$。该数据集的似然函数由下式给出：

$$\mathbf{P}(D|M)=\mathbf{P}(n_X|p_X)=\prod_X P_X^{n_X} \tag{D.25}$$

我们已经看到单一的Dirichlet分布可以作为一个自然的先验分布。然而，这样选取的先验分布，其灵活性有时会受到一些限制，特别是对所有的列或生成状态选择相同的Dirichlet分布。而Dirichlet混合分布则是一个正如参考文献［489］所述的更灵活的先验分布：

$$\mathbf{P}(P)=\sum_{i=1}^n\lambda_i\,\mathcal{D}_{\alpha_iQ_i}(P) \tag{D.26}$$

其中相同的混合又被用于所有可能的列，以反映蛋白质中氨基酸的一般性分布。混合成分项$\mathcal{D}_{\alpha_iQ_i}$的数值以及混合系数可以通过聚类方法获得。针对蛋白质模型的另一种方法是采用与PAM矩阵中列相关的向量Q_i（见第10章以及参考文献［497］）。需要指出的是，上述混合模型不同于在每一列的先验分布上应用不同的混合系数集。它也不同于以类似于第9章中介绍的HMM/NN混合模型方法，将每个P参数化为一个混合模型，以减少HMM生成参数的个数。其中，给定$n<|A|$（在参考文献［489］中，$n=9$为最优值）。我们将探索这些不同的方法留给读者作为练习。

现在我们从单一Dirichlet混合先验分布及其似然度出发，根据贝叶斯定理，可以很容易地计算出后验分布

$$\mathbf{P}(P|D)=\frac{1}{\mathbf{P}(D)}\sum_{i=1}^n\lambda_i\frac{B(\beta_i,R_i)}{B(\alpha_i,Q_i)}\mathcal{D}_{\beta_iR_i}(P) \tag{D.27}$$

新的混合项定义为

$$\beta_i=N+\alpha_i \quad 和 \quad \gamma_{iX}=\frac{n\ \ +\alpha_iq_{iX}}{N+\alpha_i} \tag{D.28}$$

β函数B定义为

$$B(\alpha,Q)=\frac{\prod_{\mathrm{X}}\Gamma(\alpha q_{\mathrm{X}})}{\Gamma(\alpha)} \tag{D.29}$$

照例有$\alpha \geqslant 0$，$q_{\mathrm{X}} \geqslant 0$和$\sum_{\mathrm{X}} q_{\mathrm{X}}=1$。共轭分布混合的后验分布仍是共轭分布的混合。在这种情况下，后验分布仍是一个Dirichlet混合分布，但其成分和混合系数不同。由于后验分布在P上的积分必为1，我们马上可以得到

$$\mathbf{P}(D)=\sum_{i=1}^{n}\lambda_i\frac{B(\beta_i,R_i)}{B(\alpha_i,Q_i)} \tag{D.30}$$

正如前面所指出的，我们无法获得MAP估计的解析表达式，尽管可以通过一些迭代过程近似地估计它。MP的估计很简单，因为它对应于后验分布的平均：

$$p_{\mathrm{X}}^{*}=\frac{1}{\mathbf{P}(D)}\sum_{i=1}^{n}\lambda_i\frac{B(\beta_i,R_i)}{B(\alpha_i,Q_i)}\gamma_{i\mathrm{X}} \tag{D.31}$$

这提供了一个在上述体系下估计模型的最优参数的公式。参考文献［489］中讨论了有关的数值计算实现问题。

D.4.2 分级Dirichlet模型

在分级模型中，我们引入一个更高层次的先验分布，例如关于上述模型的混合系数的Dirichlet先验分布。这个两层模型也是一个形如$\mathbf{P}(P|\lambda)=\sum\lambda_i\mathcal{D}_{\alpha_i Q_i}(p)$的混合模型，但它满足

$$\mathbf{P}(\lambda)=\mathcal{D}_{\beta Q}(\lambda)=\frac{\Gamma(\beta)}{\prod_i\Gamma(\beta q_i)}\prod_{i=1}^{n}\lambda_i^{\beta q_i-1} \tag{D.32}$$

于是可以得到

$$\mathbf{P}(P)=\int_{\lambda}\mathbf{P}(P|\lambda)\mathbf{P}(\lambda)d\lambda \tag{D.33}$$

交换求和与积分运算可得

$$\mathbf{P}(P)=\sum_{i=1}^{n}\mathcal{D}_{\alpha_i Q_i}(P)\left[\int_{\lambda}\lambda_i\mathcal{D}_{\beta Q}(\lambda)d\lambda\right]=\sum_{i=1}^{n}q_i\mathcal{D}_{\alpha_i Q_i}(P) \tag{D.34}$$

第二个等式由Dirichlet期望公式导出。因此这个两级分级模型实际上等价于一个一级Dirichlet混合模型，其中混合系数q_i为分级模型中第二级Dirichlet先验分布的数学期望。

附录E　高斯过程、核方法及支持向量机

本附录中将简要回顾几类重要的机器学习方法：高斯过程、核方法（kernel method）以及支持向量机。[533,141]

E.1　高斯过程模型

考虑一个回归问题，有K对从某个未知分布中提取的输入–输出训练样本（x_1, y_1），…，（x_K, y_K）。输入x是一个n维向量。为简单起见，我们假设y是一维的，但很容易将其扩展到多维的情况。回归的目标是从给定的样本中学习x和y之间的函数关系。高斯过程建模方法[559,206,399]又称为“kriging”，为回归和分类问题提供了灵活的概率体系。许多非参数回归模型等价于高斯过程，例如带有一个节点数无限的隐层且权重分布为高斯先验分布的神经网络。[398]高斯过程也可以用于直接确定函数在空间上的概率分布，而不需要以神经网络体系为基础。

高斯过程是一些变量$Y=(y(x_1), y(x_2), \cdots)$的集合，服从如下形式的联合高斯分布

$$\mathbf{P}\left(Y \middle| C, \{x_i\}\right) = \frac{1}{Z}\exp\left(-\frac{1}{2}(Y-\mu)^T C^{-1}(Y-\mu)\right) \qquad \text{(E.1)}$$

上式对任意$\{x_i\}$序列成立，其中的μ是均值向量，$C_{ij}=C(x_i, x_j)$为x_i和x_j之间的协方差。为简单起见，下面将假设$\mu=0$。对噪声和建模函数的先验假设体现在协方差矩阵中。以下面将描述各种合理确定C的参数的方法。根据（E.1），与一个测试实例x对应的变量y的预期分布可以通过观察到的训练样本得到，换言之，简

单的计算表明y服从高斯分布

$$\mathbf{P}\left(y\middle|\{y_1,\cdots,y_K\},C(x_i,x_j),\{x_1,\cdots,x_K,x\}\right)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{\left(y-y^*\right)^2}{2\sigma^2}\right) \quad (E.2)$$

其中

$$y^*=k(x)^T C_K^{-1}(y_1,\cdots,y_K) \text{ 且 } \sigma=C(x,x)-k(x)^T C_K^{-1}k(x) \quad (E.3)$$

其中$k(x)=(C(x_1,x),\cdots,C(x_K,x))$，$C_K$表示基于$K$个训练样本的协方差矩阵。

E.1.1 协方差的参数化

高斯过程模型由它的协方差函数确定，协方差函数$C(x_i,x_j)$的惟一约束是它必须对任意输入得到半正定矩阵。对于平衡信号的情况，调和分析中的Bochner定理（见参考文献［177］，完整的内容将在下文给出）利用傅里叶变换给出了这类函数的完整特征。我们已知两个正（或正定）矩阵的和也是正的（或正定的），因此协方差矩阵可以方便地参数化为各种正成分的和。有用的成分具有以下形式：

- 噪声方差：$\delta_{ij}\theta_1^2$，或者更一般的形式$\delta_{ij}f(x_i)$，后者用于与输入相关的噪声模型。
- 平滑协方差：$C(x_i,x_j)=\theta_2^2\exp\left(-\sum_{\mu=1}^n\rho_\mu^2\left(x_{iu}-x_{ju}\right)^2\right)$
- 以及更一般的形式：$C(x_i,x_j)=\theta_2^2\exp\left(-\sum_{\mu=1}^n\rho_\mu^2\left|x_{iu}-x_{ju}\right|^\gamma\right)$
- 周期协方差：$C(x_i,x_j)=\theta_3^2\exp\left(-\sum_{\mu=1}^n\rho_\mu^2\sin^2\left[\pi\left(x_{iu}-x_{ju}\right)/\gamma_u\right]\right)$

注意一个较小的ρ_u值所刻画的成分u，在很大程度上，它以一种与自动关联决定框架（automatic relevance determination framework）有关的方式与输出不相关。[398] 为简单起见，我们用θ表示模型的超参数向量。不必使用在超参数空间上冗长的蒙特卡罗积分，可以通过最小化负对数似然度

$$\mathcal{E}(\theta)=\frac{1}{2}\log\det C_K+\frac{1}{2}Y_K^T C_K^{-1}Y_K+\frac{K}{2}\log 2\pi \quad (E.4)$$

估计出一个θ值如果不用特殊的快速算法，这需要求协方差矩阵的逆，大约需要$O(N^3)$计算量。然后可以根据（E.3）实现预测或分类。例如，通过下面的方法可以立即得到两类别分类的模型：定义一个如上所述的潜在的变量Z上的高斯过程，并令

$$\mathbf{P}(y_i = 1) = \frac{1}{1+e^{-Z_i}} \tag{E.5}$$

更一般地，对于多类别的情况，可以用归一化的指数函数代替sigmoidal函数。

E.2 核方法和支持向量机

核方法和支持向量机（SVM）与高斯过程有关联，在分类和回归问题中都可以应用。简单起见，我们在这里只考虑两类别分类问题，其中给定已确知类别的样本集合（x_i, y_i），x_i是输入向量，$y_i=\pm 1$，表示对应的类别H^+或H^-。分类记号a（0, 1）与上面的表达法等价，但会导致比较累赘的记法。作为例子，下面考虑给定的蛋白质（或基因）是否属于某个家族的问题，其中给定家族内（正样本）和家族外（负样本）的氨基酸序列（或表达水平）。[275, 95] 特别地，x_i的长度可能随i变化。新样本x的类别标签y由依赖于训练样本的判别函数$\mathcal{D}(x;\{x_i, y_i\})$决定，形式是$y=\text{sign}(\mathcal{D}(x;\{x_i, y_i\}))$。用概率形式表示则是

$$y = \text{sign}\Big(\mathcal{D}\big(x;\{x_i, y_i\}\big)\Big) = \text{sign}\left(\log\frac{\mathbf{P}(H^+|x)}{\mathbf{P}(H^-|x)}\right) \tag{E.6}$$

在核方法中，判别函数用以下形式展开：

$$\mathcal{D}(x) = \sum_i y_i\lambda_i K(x_i, x) = \sum_{H^+}\lambda_i K(x_i, x) - \sum_{H^-}\lambda_i K(x_i, x) \tag{E.7}$$

因此，$\log\mathbf{P}(H^+|x) = \sum_{H^+}\lambda_j K(x_i, x)$，或者两者只相差一个无关紧要的常数，负样本也有类似关系。K称为核函数。直观上，这种处理的思想是对所有已知样本进行加权从而将新样本分类，已知样本的权重与两个因素有关：系数$\lambda_i \geqslant 0$，衡量样本i的重要程度；核$K(x_i, x)$衡量x和x_i之间的相似程度。这样判别函数的表达式直接依赖于样本，这一点和神经网络不同，后者的判别通过其已训练的参数间接依赖于训练样本。因此应用核方法时，选择合适的核函数K以及权重λ_i非常重要。不同的选择方式产生了一系列不同的方法，包括广义线性模型和SVM。

E.2.1 核函数的选择

粗略地说，根据核函数的数学理论，核函数K必须是正定的。根据泛函分析中的Mercer定理（为使内容完整，该定理将在E.3.2节给出），K可以表示为如下形式的内积：

$$K_{ij}=K(x_i, x_j)=\phi(x_i)\phi(x_j) \tag{E.8}$$

因此从另一个角度看，可以认为初始的x向量被函数$\phi(x)$映射到“特征”空间。注意特征空间的维数很高甚至为无穷；而且即使输入向量x的长度不同，特征向量$\phi(x)$的维数仍然相同。两个向量的相似性通过特征空间中的内积加以衡量。事实上也可以计算欧几里德距离$\|\phi(x_i)-\phi(x_j)\|^2=K_{ii}-2K_{ij}+K_{jj}$，它在初始的向量上定义了一种伪距离（pseudodistance）。

核方法的基本思想是在特征空间而不是在初始空间中定义一个线性或非线性的判别面。因为所有的判别都可以通过核函数和训练样本给出，所以不需要明确地构造特征空间。另外，判别面直接依赖于训练样本的一个子集——支持向量子集。

注意点积核函数提供了一种在特征空间中比较向量的方法。当它直接用于判别函数时，它对应于在特征空间中寻找线性分界超平面。使用从内积核函数K衍生出来的更复杂的核函数K'，很容易找到特征空间中更复杂的分界面（二次或更高阶次）。例如：

- 多项式核函数：$K'(x_i, x_j)=[1+K(x_i, x_j)]^m$
- 径向基核函数：$K'(x_i, x_j)=\exp\left[-\frac{1}{2\sigma^2}\left(\phi(x_i)-\phi(x_j)\right)^t\left(\phi(x_i)-\phi(x_j)\right)\right]$
- 神经网络核函数：$K'(x_i, x_j)=\tanh\left(\mu x_i^t x_j+\kappa\right)$

E.2.2 Fisher核函数

参考文献［275］给出了一种把核方法和概率生成模型结合起来的通用技术。基本的思想是：HMM之类的生成模型通常只用正样本训练，因而对判别问题来说可能并不总是最优的。然而，同时利用正、负样本及一个核函数$K(x_i, x_j)=U^t(x_i)F^{-1}U(x_j)$，可以从生成模型构造出判别模型，其中$U$是生成模型的对数似然度对于模型参数$U(x)$的梯度，$U(x)=\partial\log\mathbf{P}(x|w)/\partial w$。这个梯度描述了给定值的$w$对于生成样本$x$的贡献。对于指数族分布，梯度本质上就是充分统计量。需要再次指出，即使x的长度不同，$U(x)$仍然具有相同的长度。以在蛋白质家族上训练的HMM为例，$U(x)$是第7章中计算的导数向量。F是Fisher信息矩阵，$F=E[U(x)U^t(x)]$［对$\mathbf{P}(x|w)$取期望值］，这类核函数称为Fisher核函数。Fisher矩阵由对数似然度的2阶导数构成，因此和对应的流形（manifold）的局部曲率有关（参考文献［15］中的例子）。F确定了这个流形的黎曼度量。特别地，用两个很接近的参数w和$w+\varepsilon$ 刻画的模型，其距离是$\varepsilon^tF\varepsilon/2$。这个距离也近似于两个模型之间的相对熵。对于很多例子，至少在渐近的意义上，可以用较简单的点

积形式近似Fisher核函数。也可以用上面提到过的变换修正Fisher核函数，例如

$$K(x_i,x_j)=\exp\left[-\frac{1}{2\sigma^2}\left(U(x_j)-U(x_i)\right)^t\left(U(x_i)-U(x_j)\right)\right]$$

这说明，至少在渐近意义上，Fisher核分类器（Fisher kernel classifier）不会比生成概率模型对应的MAP判别准则差。参考文献［275］给出了一个利用Fisher核函数检测较远的同源性的例子。

E.2.3　权重选择

权重λ一般是通过迭代一个目标函数（分类损失）的优化过程而得到。通常这对应一个二次优化问题。权重常常可以看成拉格朗日算子，或者是该问题的原始参数的对偶权重（参考E.2.4节）。对于大训练集，在最优点上很多权重等于0。影响决策的只是权重不为0的向量，它们被称为支持向量。

为此考虑一个样本x_i，它的目标类别是y_i。既然我们是根据$\mathcal{D}(x_i)$的符号进行决策，那么理想情况下，我们希望$y_i\mathcal{D}(x_i)$，也就是样本i的边界能够尽可能大。由于边界随λ_i的变化也成比例变化，很自然地要对每个λ_i引入另一个约束条件$0\leq\lambda_i\leq1$。如果在特征空间中确实存在一个分界面，一个合适的目标函数是对最坏情况下的边界进行最大化。这也称为风险最小化，对应于$\max_\lambda\min_i y_i\mathcal{D}(x_i)$。SVM可以定义为一类基于结构风险最小化的核方法（参看E.2.4节）。把表达式中的$\mathcal{D}$用核函数替换，就得到$\max_\lambda\min_i\sum_j\lambda_j y_i K_{ij}$。上式可以改写为$\max_\lambda\min_i\sum_j A_{ij}\lambda_j$，其中$A_{ij}=y_iy_jK_{ij}$且$0\leq\lambda_i\leq1$。显然在最小化过程中，非零系数$A_{ij}$相对应的$\lambda_i$将是0或者1。在一个大的训练集中，很多权重都是0，在最优点处也是如此。由于现实的问题大多数无法满足边界的约束条件，我们可以采取一种相似的策略［另一种方法是使用松弛变量（slack variable），如E2.5中的例子所示］。例如，我们可以最大化边界的加权平均，权重λ_i反映了样本之间的关联程度。因此我们一般希望在λ_i满足一些线性约束条件时，最大化形如$\sum_i\lambda_i y_i\mathcal{D}(x_i)$的二次表达式。存在进行这类优化的标准技术，例如，一个用于最小化的典型函数是

$$\mathcal{E}(\lambda_i)=-\sum_i\left[y_i\lambda_i\mathcal{D}(x_i)+2\lambda_i\right]\tag{E.9}$$

这个约束最优化问题的解是惟一的，条件是对于任意有限的样本集，对应的核矩阵K_{ij}是正定的。可以用标准的迭代方法求解，但有时可能收敛得很慢。为了允许训练集中的错误和偏差，核矩阵K可以改为$K+\mu D$，其中D是对角阵，在对应正样本和负样本的位置分别为d^+和d^-。[533,108,141] 参考文献［95］中给出一个将SVM应用于基因表达数据的例子。

总之，核方法和SVM有一些很吸引人的特点。如上所述，它们是有监督学习方法，用于处理带有类别标签的数据。这些方法可以在高维空间中生成灵活的决策面。这种灵活性与核函数的选择有关。通过某种形式的边界最大化，可以控制过拟合问题。这些方法可以处理像生物序列那样的输人向量不等长的情况，还可以处理很大的特征空间。由于决策面完全用核函数和相关训练集中稀疏的支持向量子集定义，因此不需要显式地构造特征空间。学习是通过用迭代方法求解线性约束的二次型优化问题来完成的。

E.2.4 结构风险最小化和VC维数

统计学习理论给出一些不等式，用于指导包括SVM在内的一般学习系统的设计。考虑由参数向量w表示的一族分类函数 $f(x;w)$。如果数据点（x, y）是从某个联合分布$\mathbf{P}(x, y)$中得到，那么希望找到具有最小错误率或风险的函数

$$\mathcal{R}(w)=\int\frac{1}{2}|y-f(x;w)|d\mathbf{P}(x,y) \tag{E.10}$$

而风险通常是未知的，训练集上的经验风险则是已知的：

$$\mathcal{R}_K(w)=\frac{1}{2K}\sum_{1}^{K}|y_i-f(x_i;w)| \tag{E.11}$$

统计学习理论的一个基本不等式是：对于任何$0\leqslant\eta\leqslant1$，下式以概率$1-\eta$成立：

$$\mathcal{R}(w)\leqslant\mathcal{R}_K(w)+\sqrt{\frac{h(\log(2K/h)+1)-\log(\eta/4)}{K}} \tag{E.12}$$

其中h是一个称为Vapnik-Chervonenkis维数（VC维数）的非负整数。[533]
VC维数是函数集$f(x;w)$的属性。如果给定的M点的集合能够用这个函数集中的函数按所有2^M种情况分类，那么称这个点集是可以被打散的（shattered）。例如，如果$f(x;w)$是平面上的直线集合，那么所有两个点的集合都可以容易地打散，而大部分三个点的集合（除三点共线情况外）也可以被打散。但四点集合没有能被打散的。函数集$f(x;w)$的VC维数是能被打散的最大点集中的点的个数。因此，平面中所有直线的集合的VC维数为3，更一般地可以证明,在通常的n维欧几里德空间中的超平面的VC维数是$n+1$。

（E.12）基本不等式以某种方式体现了偏差/方差或者拟合/欠拟合的折中。它提示我们可以通过两种方式控制风险：经验风险（对数据的拟合程度）和学习中所用的函数集的VC维数（或者容量）。结构风险最小化的目标是通过最小化（E.

.12）右边项，而对这两项同时进行优化。

E.2.5　简单的例子：线性和广义线性模型

首先考虑$\mathcal{D}(x;w)=w_1^t x+w_2$形式的线性模型家族，其中$w=(w_1, w_2)$，$w_1$是一个向量，$w_2$是标量，它们应按比例缩放使得$\min_i|\mathcal{D}(x_i;w)|=1$。如果$R$是包含训练样本的最小球体的半径并且$\|w_1\|<A$，那么可以证明这族超平面的VC维数$h$有界：$h<R^2A^2$。这个界可能比上述的$n+1$严格得多，因此我们可以用$A$控制超平面集的容量。

如果存在一个分界超平面，那么上述比例意味着对每个样本i都有$y_i\mathcal{D}(x;w)\geqslant 1$。更一般的情形下，这个约束可能不满足，我们可以引入松弛变量$\xi_i\geqslant 0$，并且要求$y_i\mathcal{D}(x;w)\geqslant 1-\xi_i$。对（E.12）中风险的界进行最小化的支持向量方法，就是最小化

$$\mathcal{E}(w)=w^t w+\mu\sum_i \xi_i\text{，约束条件为 }\xi_1\geqslant 0\text{且}y_i\mathcal{D}(x;w)\geqslant 1-\xi_i \qquad (E.13)$$

（E.13）中的第一项产生较小的VC维，而第二项则减小的全局误差（经验风险）。引入拉格朗日算子λ_i并使用优化理论中的Kuhn-Tucker法则，就可以证明解的形式是$w=\sum_i y_i\lambda_i x_i$。这个结果也可以很直观地通过几何上的考虑得到，因为向量w都是垂直于超平面。由此得到对应于一个简单点积核函数的决策函数$\mathcal{D}(x;w)=\sum_i y_i\lambda_i x_i^t x+w_2$。只有对应于支持向量的那部分系数$\lambda_i$不为0，而且它们对应的松弛变量是饱和的：$y_i\mathcal{D}(x_i;w)=1-\xi_i$。可以通过对二次型目标函数最小化得到系数$\lambda_i$：

$$\mathcal{E}(\lambda)=-\sum_i\lambda_i+\frac{1}{2}\sum_{ij}y_i y_j\lambda_i\lambda_j x_i^t x_j\text{，约束条件为}0\leqslant\lambda_i\leqslant\mu\text{且}\sum_i\lambda_i y_i \qquad (E.14)$$

在logistic线性模型中，$\mathbf{P}(y)=\mathcal{D}(x)=\sigma(yw^t x)$，其中$w$是参数向量，$\sigma$是sigmoidal函数$\sigma(u)=1/(1+e^{-u})$。$w$的一种标准的先验分布是平均值为0、方差为$C$的高斯分布。忽略常数项后，训练集的负对数后验概率是

$$\mathcal{E}(w)=-\sum_i\log\sigma\left(y_i w^t x_i\right)+\frac{1}{2}w^t C^{-1}w \qquad (E.15)$$

容易验证最优解必须满足

$$w^*=-\sum_i y_i\lambda_i C x_i \qquad (E.16)$$

其中，$\lambda_i=\partial\log\sigma(z)/\partial z, z=y_i w^{*t}x_i$。因此我们得到了（E.7）形式的通解，核为

$K(x_i, x_j) = x_i^t C x_j$。

E.3 高斯过程和SVM的定理

为完整起见，这里将叙述两个有用的定理，它们是核方法、SVM以及高斯过程的理论基础：概率论和调和分析中的Bochner定理以及泛函分析中的Mercer定理。

E.3.1 Bochner定理

Bochner定理以傅里叶变换的形式完整地描述了特征函数的特点，并且还附带建立了连续平稳过程的特征函数和协方差函数之间的等价性。

考虑一个复随机过程，即一族复随机变量$\{X_t=U_t+iV_t\}$，其中$-\infty<t<+\infty$。为简单起见，假定$E(X_t)=0$，并用$\mathbf{Cov}(X_u, X_v)=E(X_u, \overline{X}_v)$定义协方差。我们假定过程$X_t$是平稳的和连续的，这隐含着协方差函数是连续的并且满足

$$\mathbf{Cov}(X_s, X_{s+t})=f(t) \tag{E.17}$$

因此它仅仅依赖于变量之间的距离。在这些假设条件下，Bochner定理断言f满足

$$f(t)=\int_{-\infty}^{+\infty} e^{i\lambda t}\mu(d\lambda) \tag{E.18}$$

其中μ是实数轴上的测度，总积分为$f(0)$。这意味着f是正定的，并且是一个有限测度的傅里叶变换。如果X_t是实变量，则测度μ是对称的，且有

$$f(t)=\int_{-\infty}^{+\infty} \cos\lambda t \mu(d\lambda) \tag{E.19}$$

测度μ称为过程的谱测度（spectral measure）。相反地，给定实数轴上的任意有限测度μ，可以证明存在一个平稳的随机过程X_t，其谱测度为μ。测度$\mu/f(0)$是一个概率测度，因此（E.18）中的函数f是一个特征函数。换言之，等价的定理是：连续函数$g(t)$是某一概率分布的特征函数，当且仅当它是正定的［即满足类似（E.18）的关系］并满足归一化条件$g(0)=1$。因此连续的特征函数与一个平稳过程的协方差函数是等价的，至多相差常数因子。更多细节见参考文献［177］。

E.3.2 Mercer定理

Mercer定理揭示了对称正定核函数与“特征空间”中的点积的关系。考虑两

个L_2（平方可积）空间之间的积分算子$\kappa: L_2 \to L_2$，核K是连续对称的，则

$$(\kappa f)y = \int K(x,y)f(x)dx \tag{E.20}$$

假设K也是正定的，亦即若$f \neq 0$，有

$$\int f(x)K(x,y)f(y)dxdy > 0 \tag{E.21}$$

于是存在标准正交的函数基ξ_i（x），使得K可以被表示为如下形式：

$$K(x,y) = \sum_{i=1}^{\infty} \lambda_i \xi_i(x)\xi_i(y) \tag{E.22}$$

其中$\lambda_i \geqslant 0$，并且任意一对i和j的标量积（$\xi_i \xi_j$）$_{L_2}=\delta_{ij}$（标准正交）。根据（E.20）和标准正交条件，我们可以得到

$$(\kappa \xi_i)y = \int \sum_{j=1}^{\infty} \lambda_j \xi_j(x)\xi_j(y)\xi_i(x)dx = \lambda_i \xi_i(y) \tag{E.23}$$

换言之，κ是一个紧算子，可以进行特征分解，其特征向量为ξ_i，特征值为非负值λ_i。如果我们定义函数$\phi(x)$为

$$\phi(x) = \sum_{i=1}^{\infty} \sqrt{\lambda_i}\,\xi_i(x) \tag{E.24}$$

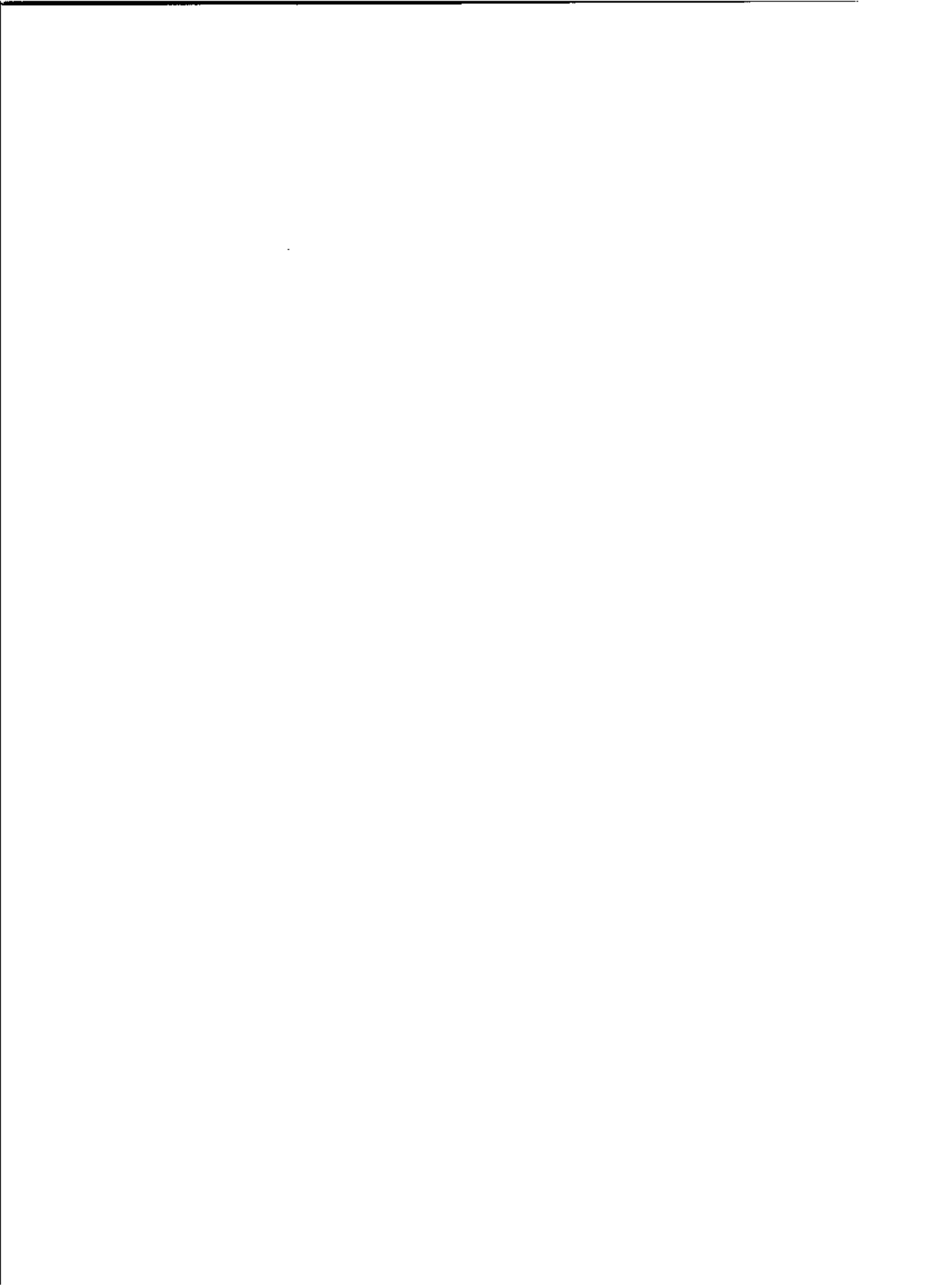
则再次使用标准正交条件可以得到

$$K(x,y) = \phi(x)\phi(y) \tag{E.25}$$

它正是（E.8）中需要的分解。反之，如果我们使用连续的$\phi(x)$把x嵌入M维的特征空间，我们就可以用（E.25）定义一个连续的核$K(x, y)$。对应的算子是正定的，因为

$$\int f(x)K(x,y)f(y)dxdy = \int f(x)\big(\phi(x)\phi(y)\big)f(y)dxdy =$$

$$\sum_{i=1}^{M}\int f(x)\phi_i(x)\phi_i(y)f(y)dxdy = \sum_{i=1}^{M}\left(\int f(x)\phi_i(x)dx\right)^2 \geqslant 0 \tag{E.26}$$



附录F　公式和缩写符号

概率论

- π：未标定置信度
- $\mathbf{P}(P, Q, R, \cdots)$：概率（真实概率分布）
- $\mathbf{E}(\mathbf{E}_Q)$：期望（对概率分布Q求期望）
- **Var**：方差
- **Cov**：协方差
- $X_i, Y_i\ (x_i, y_i)$：命题或随机变量（其中x_i是X_i的真实值）
- $\overline{X}$：集合X的补集或命题X的否命题
- $X \perp Y\ (X \perp Y \mid Z)$：随机变量$X$和$Y$独立（给定$Z$时条件独立）
- $\mathbf{P}(x_1,\cdots,x_n)$：事件$X_1 = x_1,\cdots,X_n=x_n$的概率。在上下文确定的情形中，也可以写成$\mathbf{P}(X_1,\cdots, X_n)$。同样地，对于特定的密度函数$Q$，写成$Q(x_1,\cdots,x_n)$或$Q(X_1, \cdots, X_n)$
- $\mathbf{P}(X|Y)\ (\mathbf{E}(X|Y))$：条件概率（条件期望）
- $\mathcal{N}(\mu,\sigma)$，$\mathcal{N}(\mu,C)$，$\mathcal{N}(\mu,\sigma^2)$，$N(x;\mu,\sigma^2)$：正态（高斯）分布，其均值为$\mu$、方差为$\sigma^2$或协方差矩阵为$C$
- $\Gamma(w|\alpha, \lambda)$：具有参数$\alpha$和$\lambda$的伽玛密度
- $\mathcal{D}_{\alpha Q}$：具有参数α和Q的Dirichlet分布
- $t(x; v, m, \sigma^2)$，$t(v, m, \sigma^2)$：自由度为v、位置为m、尺度因子为σ的学生氏分布
- $\mathcal{I}(x; v, \sigma^2)$，$\mathcal{I}(v, \sigma^2)$：自由度为$v$、尺度因子为$\sigma$的标定逆伽玛分布

函 数

- $\mathcal{E}$：能量、误差、负对数似然度或负对数后验概率（根据上下文决定是何种含义）
- $\mathcal{E}_T, \mathcal{E}_G, \mathcal{E}_C$：训练误差、推广误差、分类误差
- $\mathcal{E}_P$：吝啬误差
- $\mathcal{F}$：自由能
- $\mathcal{L}$：拉格朗日算子
- $\mathcal{D}$：决策函数
- $\mathcal{R}$：风险函数
- $\mathcal{R}_K$：经验风险函数
- $\mathcal{H}(P)$，$\mathcal{H}(X)$：分布P或随机变量X的熵，考察随机变量时指微分熵
- $\mathcal{H}(P, Q)$，$\mathcal{H}(X, Y)$：分布P与Q（或随机变量X与Y）的相对熵
- $\mathcal{I}(P, Q)$，$\mathcal{I}(X, Y)$：分布P与Q（或随机变量X与Y）之间的互信息
- Z：分割函数或归一化因子（有时也用C）
- C：常数或归一化因子
- $\delta(x, y)$：Kronecker函数，当x=y时取值为1，其余情况取值为0
- f, f'：函数f及其导数
- $\Gamma(x)$：伽玛函数
- $B(\alpha, Q)$：贝塔函数（附录D）
- 我们也使用∪代表上凸（2阶导数为正），∩代表下凸（2阶导数为负），而不使用容易引起混淆的“凸”、“凹”概念

模型、字符集和序列

- M（M=M（w））：（具有参数w的）模型
- D：数据
- I：背景信息
- H：隐变量（原因）
- $S=\{s_1, s_2, \cdots, s_{|S|}\}$：系统的状态集
- s：状态
- A(X)：字符集
- $A=\{$A, C, G, T$\}$：DNA字符集
- $A=\{$A, C, G, U$\}$：RNA字符集

- $A=\{\mathrm{A, C, D}, \cdots\}$：氨基酸字符集
- A^*：由A中字符组成的所有有限长的串所构成的集合
- $O=(X^1 \cdots X^t \cdots)$：序列（其中“$O$”表示“观察量”或“排序的”）
- $\varnothing$：空序列
- $O_1, \cdots, O_K$：训练序列集
- O_k^j：第k条序列的第j个字符

图和集合

- $G=(V, E)$：顶点集为V、边集为E的无向图
- $G=(V, \vec{E})$：顶点集为V、边集为E的有向图
- T：树
- $N(i)$：顶点i的邻节点
- $N^+(i)$：有向图中顶点i的子节点
- $N^-(i)$：有向图中顶点i的父节点
- $C^+(i)$：有向图中顶点i的后代
- $C^-(i)$：有向图中顶点i的祖先
- $N(I)$：顶点集I的邻节点（或边界）
- $\mathcal{P}(G)$：满足图G所确定的条件独立性假设的概率分布族
- G^C：图G的团图
- G^M:图G的伦图
- $\cup, \cap, ^-$：集合的交、并和补运算
- $\varnothing$：空集

维　数

- $|A|$：字符集符号数
- $|S|$：状态数
- $|H|$：HMM/NN混合模型中的隐节点数
- N：序列的长度（平均长度）
- K：序列或样本的数目（例如在训练集中）
- T：时间标度（在不产生混淆的情况下也指温度）

一般参数

- w：参数向量

- t_{ji}：从状态i到j的转移概率，例如在马尔可夫链中
- $t\left(w_{ij}^{t}, \mathrm{X}^{t}\right)$:算法迭代过程或序列中的时间指标
- $^{+},{}^{-}\left(w_{ij}^{+}\right)$：算法迭代过程中的相对时间指标
- $^{*}\left(w_{ij}^{*}\right)$：最优解
- η：学习率

神经网络

- w_{ij}：从节点j到i的连接权重
- w_i, λ_i：节点i的偏差、增益
- $D_j=(d_j, t_j)$：训练样本，d_j为输入向量，t_j为相应的目标输出向量
- $y_i=f_i(x_i)$：节点i的输入–输出关系，x_i为节点的总输入，f_i为激活函数，y_i为输出
- $y(d_i)$：神经网络输入向量为d_i时的输出
- $y_j(d_i)$：神经网络输入为d_i时，第j个输出节点的输出
- $t_j(d_i)$：神经网络输入为d_i时，第j个输出节点的目标输出值

隐马氏模型

- m, d, i, h：主干、删除、插入和定位状态；对于大部分情况，i只是一个下标
- $start, end$：HMM的起始和终止状态（在图中也写成S和E）
- E：模型的生成状态集合
- D：模型的删除（哑）状态集合
- L：仅在附录D中出现，L表示HMM环状构架中的状态集
- $t_{ij}(w_{ij})$：从状态j到i的转移概率（归一化指数函数表示）
- $e_{iX}(w_{iX})$：状态i生成字符X的生成概率（归一化指数函数表示）
- t_{ij}^{D}：从状态j到i的哑转移概率
- π：路径变量
- $n(i, \mathrm{X}, \pi, O)$：对于给定HMM中的序列O，状态i沿路径π生成字符X的次数
- $\alpha_i(t)$：前向变量
- $\alpha_i^{L}(t)$：HMM环状构架中的前向变量
- $\beta_i(t)$：后向变量
- $\hat{\alpha}_i^{L}(t)$：标定的前向变量
- $\hat{\beta}_i(t)$：标定的后向变量
- $\lambda_i(t)$：对于HMM中的一个给定观察序列，t时刻处于状态i的概率

- $\gamma_{ji}(t)$：对于HMM中的一个给定观察序列，t时刻由状态i转移到j的概率
- $\delta_i(t)$：Viterbi算法递归过程中使用的变量
- κ：HMM哑循环的概率
- b（X, Y, Z）：三元组XYZ的可弯曲性
- $B(i, O)$：序列O上i位置的可弯曲性
- $B(i)$：一族序列中i位置的可弯曲性
- W：用于可弯曲性计算的平均窗长度

双向结构

- W：参数总个数
- O_t：输出的概率向量
- B_t：后向上下文向量
- F_t：前向上下文向量
- I_t：输入向量
- $\eta(.)$：输出函数
- $\beta(.)$：后向转移函数
- $\phi(.)$：前后转移函数
- n：链中状态数
- q：平移算子

文法

- L：语言
- G：文法
- $L(G)$：由文法G生成的语言
- R：文法的产生式规则
- V：变量字符集
- $s = start$：起始变量
- $\alpha\to\beta$：文法的产生式规则：α“产生”或“扩展成”β
- $\pi_i(t)$：文法中的衍生变量
- $n(\beta, u, \pi, O)$：从给定文法中推导出关于序列O的结论π，需要应用产生式规则$u\to\beta$的次数
- $P_{\alpha\to\beta}(w_{\alpha\to\beta})$：随机文法中产生式规则$\alpha\to\beta$的概率（用归一化指数函数表示）

系统进化树

- r：根节点
- X_i：与顶点i相关联的字符
- d_{ji}：节点i与j之间的时间距离
- $p_{X_jX_i}$（d_{ji}）:经过时间d_{ji}，X_i被X_j替换的概率
- $\chi^i(t)$：与t时刻序列位置i上的字符相关的随机变量
- $p^i_{YX}(t)$：经过时间t，序列位置i上的字符X被Y替换的概率
- $P(t)=(p_{YX}(t))$：t时刻的替换概率矩阵
- $Q=(q_{YX})$：0时刻矩阵P的导数［$Q=P'(0)$］
- $p=(p_X)$：稳态分布
- χ_i：与树的节点i相关的随机变量
- I：树的内节点集
- $O^+(i)$：以节点i为根节点的子树所包含的证据

微阵列

- n（n_c, n_t）：（处理、对照实验中）一个基因表达水平的测量次数
- $x_1^c,\cdots,x_{n_c}^c\left(x_1^t,\cdots,x_{n_t}^t\right)$：对照（处理）实验中，一个基因表达水平的测量次数
- m（m_c, m_t）：（处理、对照实验中）一个基因的平均表达水平
- $s^2\left(s_c^2, s_t^2\right)$：（处理、对照实验中）一个基因表达水平的方差
- $d_1,\cdots,d_N$：进行聚类的N个数据点
- K：聚类类别数

核方法和支持向量机

- w：模型参数向量
- λ_i：权重
- ξ_i：松弛变量
- $K_{ij}=K(x_i,x_j)$：核函数
- F：Fisher信息矩阵
- $\phi(x)$：特征向量
- $U(x)$：对数似然度关于模型参数的梯度向量
- h：VC维数

缩写符号

- CFG：context-free grammar，上下文无关文法
- CSG：context-sensitive grammar，上下文相关文法
- BIOHMM：bidirectional IOHMM，双向IOHMM
- BRNN：bidirectional RNN，双向RNN
- EM：expectation maximization，期望最大化
- HMM：hidden Markov model，隐马氏模型
- IOHMM：input-output HMM，输入–输出HMM
- LMS：least mean square，最小均方
- MAP：maximum a posteriori，最大后验分布
- MaxEnt：maximum entropy，最大熵
- MCMC：Markov chain Monte Carlo，马尔可夫链—蒙特卡罗方法
- ML：maximum likelihood，最大似然估计
- MLP：multilayer perceptron，多层感知器
- MP：mean posterior，平均后验分布
- NN：neural network，神经网络
- RNN：recursive NN，反馈型神经网络
- RG：regular grammar，正则文法
- REG：recursively enumerable grammar，递归可枚举文法
- SG：stochastic grammar，随机文法
- SCFG：stochastic context-free grammar，随机上下文无关文法
- SS：secondary structure，二级结构
- SVM：support vector machine，支持向量机
- VC：Vapnik-Chervonenkis

参考文献

[1] Y. Abu-Mustafa. Machines that learn from hints. *Sci. American*, 272:64–69, 1995.

[2] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski. A learning algorithm for Boltzmann machines. *Cognitive Science*, 9:147–169, 1985.

[3] A. V. Aho, R. Sethi, and J. D. Ullman. *Compilers. Principles, Techniques, and Tools.* Addison-Wesley, Reading, MA, 1986.

[4] S. M. Aji and R. J. McEliece. The generalized distributive law. Technical Report, Department of Electrical Engineering, California Institute of Technology, 1997.

[5] H. Akaike. A new look at the statistical model identification. *IEEE Trans. Aut. Control*, 19:716–723, 1974.

[6] C. Alff-Steinberger. The genetic code and error transmission. *Proc. Natl. Acad. Sci. USA*, 64:584–591, 1969.

[7] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA*, 96:6745–6750, 1999.

[8] S. F. Altschul. Amino acid substitution matrices from an information theoretic perspective. *J. Mol. Biol.*, 219:555–565, 1991.

[9] S. F. Altschul, M. S. Boguski, W. Gish, and J. C. Wootton. Issues in searching molecular sequence databases. *Nat. Genet.*, 6:119–129, 1994.

[10] S. F. Altschul, R. Carrol, and D. J. Lipman. Weights for data related by a tree. *J. Mol. Biol.*, 207:647–653, 1989.

[11] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215:403–410, 1990.

[12] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and L. J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucl. Acids Res.*, 25:3389–3402, 1997.

[13] S.F. Altschul. A protein alignment scoring system sensitive at all evolutionary

distances. *J. Mol. Evol.*, 36:290-300, 1993.

[14] S.F. Altschul. Local alignment statistics. *Meth. Enzymol.*, 274:460-480, 1996.

[15] S. Amari. Natural gradient works efficiently in learning. *Neural Comp.*, 10:251-276, 1998.

[16] S. Amari and N. Murata. Statistical theory of learning curves under entropic loss criterion. *Neural Comp.*, 5:140-153, 1993.

[17] C. A. F. Andersen and S. Brunak. Amino acid subalphabets can improve protein structure prediction. *Submitted*, 2001.

[18] M. A. Andrade, G. Casari, C. Sander, and A. Valencia. Classification of protein families and detection of the determinant residues with an improved self-organizing map. *Biol. Cybern.*, 76:441-450, 1997.

[19] S. M. Arfin, A. D. Long, E. T. Ito, L. Tolleri, M. M. Riehle, E. S. Paegle, and G. W. Hatfield. Global gene expression profiling in *escherichia coli* K12: the effects of integration host factor. *J. Biol. Chem.*, 275:29672-29684, 2000.

[20] B. C. Arnold and S. J. Press. Compatible conditional distributions. *J. Amer. Statist. Assn.*, 84:152-156, 1989.

[21] M. Ashburner. On the representation of gene function in genetic databases. *ISMB*, 6, 1998.

[22] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. *Nature Genet.*, 25:25-29, 2000.

[23] A. Bairoch. The PROSITE dictionary of sites and patterns in proteins, its current status. *Nucl. Acids Res.*, 21:3097-3103, 1993.

[24] A. Bairoch and R. Apweiler. The SWISS-PROT protein sequence data bank and its supplement TrEMBL. *Nucl. Acids Res.*, 25:31-36, 1997.

[25] J. K. Baker. Trainable grammars for speech recognition. In J. J. Wolf and D. H. Klat, editors, *Speech Communication Papers for the 97th Meeting of the Acoustical Society of America*, pages 547-550, 1979.

[26] P. Baldi. Gradient descent learning algorithms overview: A general dynamical systems perspective. *IEEE Trans. on Neural Networks*, 6:182-195, 1995.

[27] P. Baldi. Substitution matrices and hidden Markov models. *J. Comput. Biol.*, 2:497-501, 1995.

[28] P. Baldi. On the convergence of a clustering algorithm for protein-coding regions in microbial genomes. *Bioinformatics*, 16:367-371, 2000.

[29] P. Baldi. *The Shattered Self-the End of Natural Evolution.* MIT Press, Cambridge, MA, 2001.

[30] P. Baldi and Pierre-Francois Baisnee. Sequence analysis by additive scales: DNA structure for sequences and repeats of all lengths. *Bioinformatics*, 16:865-889, 2000.

[31] P. Baldi, S. Brunak, Y. Chauvin, C. A. F. Andersen, and H. Nielsen. Assessing the

accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16:412-424, 2000.

[32] P. Baldi, S. Brunak, Y. Chauvin, J. Engelbrecht, and A. Krogh. Hidden Markov models for human genes. In G. Tesauro J. D. Cowan and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 6, pages 761-768. Morgan Kaufmann, San Francisco, 1994.

[33] P. Baldi, S. Brunak, Y. Chauvin, J. Engelbrecht, and A. Krogh. Periodic sequence patterns in human exons. In *Proceedings of the 1995 Conference on Intelligent Systems for Molecular Biology (ISMB95)*. AAAI Press, Menlo Park, CA, 1995.

[34] P. Baldi, S. Brunak, Y. Chauvin, and A. Krogh. Naturally occurring nucleosome positioning signals in human exons and introns. *J. Mol. Biol.*, 263:503-510, 1996.

[35] P. Baldi, S. Brunak, Y. Chauvin, and A. Krogh. Hidden Markov models for human genes: periodic patterns in exon sequences. In S. Suhai, editor, *Theoretical and Computational Methods in Genome Research*, pages 15-32, New York, 1997. Plenum Press.

[36] P. Baldi, S. Brunak, P. Frasconi, G. Pollastri, and G. Soda. Bidirectional dynamics for protein secondary structure prediction. In R. Sun and C. L. Giles, editors, *Sequence Learning: Paradigms, Algorithms, and Applications*, pages 99-120. Springer Verlag, New York, 2000.

[37] P. Baldi, S. Brunak, P. Frasconi, G. Soda, and G. Pollastri. Exploiting the past and the future in protein secondary structure prediction. *Bioinformatics*, 15:937-946, 1999.

[38] P. Baldi and Y. Chauvin. Hidden Markov models of the G-protein-coupled receptor family. *J. Comput. Biol.*, 1:311-335, 1994.

[39] P. Baldi and Y. Chauvin. Smooth on-line learning algorithms for hidden Markov models. *Neural Comp.*, 6:305-316, 1994.

[40] P. Baldi and Y. Chauvin. Hybrid modeling, HMM/NN architectures, and protein applications. *Neural Comp.*, 8:1541-1565, 1996.

[41] P. Baldi, Y. Chauvin, T. Hunkapillar, and M. McClure. Hidden Markov models of biological primary sequence information. *Proc. Natl. Acad. Sci. USA*, 91:1059-1063, 1994.

[42] P. Baldi, Y. Chauvin, F. Tobin, and A. Williams. Mining data bases of partial protein sequences using hidden Markov models. Net-ID/SmithKline Beecham Technical Report, 1996.

[43] P. Baldi and G. Wesley Hatfield. *Microarrays and Gene Expression*. Cambridge University Press, Cambridge, UK, 2001.

[44] P. Baldi and A. D. Long. A Bayesian framework for the analysis of microarray expression data: regularized t-test and statistical inferences of gene changes. *Bioinformatics*, 17(6):509-519, 2001.

[45] P. Baldi, G. Pollastri, C. A. F. Andersen, and S. Brunak. Matching protein β-sheet partners by feedforward and recurrent neural networks. In *Proceedings of the 2000 Conference on Intelligent Systems for Molecular Biology (ISMB00), La Jolla, CA*, pages 25-36. AAAI Press, Menlo Park, CA, 2000.

[46] P. Baldi, G. Pollastri, C. A. F. Andersen, and S. Brunak. Matching protein beta-

sheet partners by feedforward and recurrent neural networks. *ISMB*, 8:25-36, 2000.

[47] J. M. Baldwin. The probable arrangement of the helices in G protein coupled receptors. *EMBO J.*, 12:1693-1703, 1993.

[48] F. G. Ball and J. A. Rice. Stochastic models for ion channels: Introduction and bibliography. *Mathemat. Biosci.*, 112:189-206, 1992.

[49] N. Barkai, H. S. Seung, and H. Sompolinsky. Local and global convergence of online learning. *Phys. Rev. L*, 75:1415-1418, 1995.

[50] N. Barkai and H. Sompolinsky. Statistical mechanics of the maximum-likelihood density-estimation. *Phys. Rev. E*, 50:1766-1769, 1994.

[51] V. Barnett. *Comparative Statistical Inference.* John Wiley, New York, 1982.

[52] G. J. Barton. Protein multiple sequence alignment and flexible pattern matching. *Meth. Enzymol.*, 183:403-427, 1990.

[53] E. B. Baum. Toward a model of mind as a laissez-faire economy of idiots. Preprint, 1997.

[54] L. E. Baum. An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. *Inequalities*, 3:1-8, 1972.

[55] A. A. Beaudry and G. F. Joyce. Directed evolution of an RNA enzyme. *Science*, 257:635-641, 1992.

[56] T. C. Bell, J. G. Cleary, and I. H. Witten. *Text Compression.* Prentice-Hall, Englewood Cliffs, NJ, 1990.

[57] Y. Bengio, Y. Le Cunn, and D. Henderson. Globally trained handwritten word recognizer using spatial representation, convolutional neural networks and hidden Markov models. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 6, pages 937-944. Morgan Kaufmann, San Francisco, CA, 1994.

[58] Y. Bengio and P. Frasconi. An input-output HMM architecture. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 427-434. Morgan Kaufmann, San Francisco, 1995.

[59] R. Benne. RNA editing. The long and the short of it. *Nature*, 380:391-392, 1996.

[60] S. A. Benner. Patterns of divergence in homologous proteins as indicators of tertiary and quaternary structure. *Adv. Enzyme Regul.*, 28:219-236, 1989.

[61] S. A. Benner. Predicting the conformation of proteins from sequences. Progress and future progress. *J. Mol. Recog.*, 8:9-28, 1995.

[62] D. A. Benson, M. S. Boguski, D. J. Lipman, and J. Ostell. GenBank. *Nucl. Acids Res.*, 25:1-6, 1997.

[63] J. O. Berger. *Statistical Decision Theory and Bayesian Analysis.* Springer-Verlag, New York, 1985.

[64] A. L. Berman, E. Kolker, and E. N. Trifonov. Underlying order in protein sequence organization. *Proc. Natl. Acad. Sci. USA.*, 91:4044-4047, 1994.

[65] G. Bernardi. The human genome: Organization and evolutionary history. *Ann. Rev. Genetics*, 29:445-476, 1995.

[66] D. Bertsekas. *Dynamic Programming and Optimal Control.* Athena Scientific, Belmont, MA, 1995.

[67] D. Bertsimas and J. Tsitsiklis. Simulated annealing. *Statis. Sci.*, 8:10-15, 1993.

[68] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *J. R. Statis. Soc. B*, 36:192-225, 1974.

[69] J. Besag, P. Green, D. Higdon, and K. Mengersen. Bayesian computation and stochastic systems. *Statis. Sci.*, 10:3-66, 1995.

[70] C. M. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, 1995.

[71] R. E. Blahut. *Principles and Practice of Information Theory.* Addison-Wesley, Reading, MA, 1987.

[72] M. Blatt, S. Wiseman, and E. Domany. Super-paramagnetic clustering of data. *Phys. Review Lett.*, 76:3251-3254, 1996.

[73] G. Blobel. Intracellular membrane topogenesis. *Proc. Natl. Acad. Sci. USA*, 77:1496, 1980.

[74] N. Blom, S. Gammeltoft, and S. Brunak. Sequence and structure-based prediction of eukaryotic protein phosphorylation sites. *J. Mol. Biol.*, 294:1351-1362, 1999.

[75] N. Blom, J. Hansen, D. Blaas, and S. Brunak. Cleavage site analysis in picornaviral polyproteins by neural networks. *Protein Sci.*, 5:2203-2216, 1996.

[76] M. Bloom and O. G. Mouritsen. The evolution of membranes. In R. Lipowsky and E. Sackmann, editors, *Handbook of Biological Physics vol. 1*, pages 65-95, Amsterdam, 1995. Elsevier Science.

[77] G. Bohm. New approaches in molecular structure prediction. *Biophys. Chem.*, 59:1-32, 1996.

[78] H. Bohr, J. Bohr, S. Brunak, R. M. J. Cotterill, B. Lautrup, L. Nørskov, O. H. Olsen, and S. B. Petersen. Protein secondary structures and homology by neural networks: The α-helices in rhodopsin. *FEBS Letters*, 241:223-228, 1988.

[79] P. Bork, C. Ouzounis, and C. Sander. From genome sequences to protein function. *Curr. Opin. Struct. Biol.*, 4:393-403, 1994.

[80] P. Bork, C. Ouzounis, C. Sander, M. Scharf, R. Schneider, and E. Sonnhammer. Comprehensive sequence analysis of the 182 predicted open reading frames of yeast chromosome iii. *Protein Sci.*, 1:1677-1690, 1992.

[81] M. Borodovsky and J. McIninch. Genmark: Parallel gene recognition for both DNA strands. *Computers Chem.*, 17:123-133, 1993.

[82] M. Borodovsky, J. D. McIninch, E. V. Koonin, K. E. Rudd, C. Medigue, and A. Danchin. Detection of new genes in a bacterial genome using Markov models for three gene classes. *Nucl. Acids Res.*, 23:3554-3562, 1995.

[83] M. Borodovsky, K. E. Rudd, and E. V. Koonin. Intrinsic and extrinsic approaches for detecting genes in a bacterial genome. *Nucl. Acids Res.*, 22:4756-4767, 1994.

[84] H. Bourlard and N. Morgan. *Connectionist Speech Recognition: A Hybrid Approach.* Kluwer Academic, Boston, 1994.

[85] J. M. Bower and D. Beeman. *The Book of Genesis: Exploring Realistic Neural Models with the GEneral NEural SImulations System.* Telos/Springer-Verlag, New

York, 1995.

[86] G. E. P. Box and G. C. Tiao. *Bayesian Inference in Statistical Analysis.* Addison-Wesley, Reading, MA, 1973.

[87] A. Brack and L. E. Orgel. Beta structures of alternating polypeptides and their possible prebiotic significance. *Nature*, 256:383–387, 1975.

[88] D. Bray. Protein molecules as computational elements in living cells. *Nature*, 376:307–312, 1995.

[89] A. Brazma, I. J. Jonassen, J. Vilo, and E. Ukkonen. Predicting gene regulatory elements in silico on a genomic scale. *Genome Res.*, 8:1202–1215, 1998.

[90] L. Breiman. Discussion of neural networks and related methods for classification. *J. R. Statis. Soc. B*, 56:409–456, 1994.

[91] V. Brendel and H. G. Busse. Genome structure described by formal languages. *Nucl. Acids Res.*, 12:2561–2568, 1984.

[92] S. Brenner, G. Elgar, R. Sandford, A. Macrae, B. Venkatesh, and S. Aparicio. Characterization of the pufferfish (Fugu) genome as a compact model vertebrate genome. *Nature*, 366:265–268, 1993.

[93] S. E. Brenner, C. Chothia, and T. J. P. Hubbard. Population statistics of protein structures: lessons from structural classification. *Curr. Opin. Struct. Biol.*, 7:369–376, 1997.

[94] L. D. Brown. *Fundamentals of Statistical Exponential Families.* Institute of Mathematical Statistics, Hayward, CA, 1986.

[95] M. P. S. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. Walsh Sugnet, T. S. Furey, M. Ares, and D. Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc. Natl. Acad. Sci. USA*, 97:262–267, 2000.

[96] I. Brukner, R. Sánchez, D. Suck, and S. Pongor. Sequence-dependent bending propensity of DNA as revealed by DNase I: parameters for trinucleotides. *EMBO J.*, 14:1812–1818, 1995.

[97] S. Brunak. Non-linearities in training sets identified by inspecting the order in which neural networks learn. In O. Benhar, C. Bosio, P. Del Giudice, and E. Tabet, editors, *Neural Networks: From Biology to High Energy Physics*, pages 277–288, Pisa, 1991. ETS Editrice.

[98] S. Brunak. Doing sequence analysis by inspecting the order in which neural networks learn. In D. M. Soumpasis and T. M. Jovin, editors, *Computation of Biomolecular Structures — Achievements, Problems and Perspectives*, pages 43–54, Berlin, 1993. Springer–Verlag.

[99] S. Brunak and J. Engelbrecht. Correlation between protein secondary structure and the mRNA nucleotide sequence. *Proteins*, 25:237–252, 1996.

[100] S. Brunak, J. Engelbrecht, and S. Knudsen. Cleaning up gene databases. *Nature*, 343:123, 1990.

[101] S. Brunak, J. Engelbrecht, and S. Knudsen. Neural network detects errors in the assignment of pre-mRNA splice site. *Nucl. Acids Res.*, 18:4797–4801, 1990.

[102] S. Brunak, J. Engelbrecht, and S. Knudsen. Prediction of human mRNA donor

and acceptor sites from the DNA sequence. *J. Mol. Biol.*, 220:49–65, 1991.

[103] S. Brunak and B. Lautrup. *Neural Networks—Computers with Intuition.* World Scientific Pub., Singapore, 1990.

[104] J. Buhmann and H. Kuhnel. Vector quantization with complexity costs. *IEEE Trans. Information Theory*, 39:1133–1145, 1993.

[105] C. J. Bult, O. White, G. J. Olsen, L. Zhou, R. D. Fleischmann, G. G. Sutton, J. A. Blake, L. M. FitzGerald, R. A. Clayton, J. D. Gocayne, A. R. Kerlavage, B. A. Dougherty, J. F. Tomb, M. D. Adams, C. I. Reich, R. Overbeek, E. F. Kirkness, K. G. Weinstock, J. M. Merrick, A. Glodek, J. L. Scott, N. S. M. Geoghagen, and J. C. Venter. Complete genome sequence of the methanogenic archaeon, *Methanococcus jannaschii. Science*, 273:1058–1073, 1996.

[106] W. Buntine. A guide to the literature on learning probabilistic networks from data. *IEEE Trans. Knowledge Data Eng.*, 8:195–210, 1996.

[107] C. Burge and S. Karlin. Prediction of complete gene structures in human genomic dna. *J. Mol. Biol.*, 268:78–94, 1997.

[108] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[109] J. M. Burke and A. Berzal-Herranz. In vitro selection and evolution of RNA: Applications for catalytic RNA, molecular recognition, and drug discovery. *Faseb J.*, 7:106–112, 1993.

[110] M. Burset and R. Guigó. Evaluation of gene structure prediction programs. *Genomics*, 34:353–367, 1996.

[111] H. J. Bussemaker, H. Li, and E. D. Siggia. Building a dictionary for genomes: identification of presumptive regulatory sites by statistical analysis. *Proc. Natl. Acad. Sci. USA*, 97:10096–10100, 2000.

[112] C. R. Calladine and H. R. Drew. *Understanding DNA—The Molecule and How it Works.* Academic Press, London, 1992.

[113] L. R. Cardon and G. D. Stormo. Expectation-maximization algorithm for identifying protein-binding sites with variable lengths from unaligned DNA fragments. *J. Mol. Biol.*, 223:159–170, 1992.

[114] C. R. Carlson and A. B. Kolsto. A small (2.4 mb) bacillus cereus chromosome corresponds to a conserved region of a larger (5.3 mb) bacillus cereus chromosome. *Mol. Microbiol.*, 13:161–169, 1994.

[115] R. Caruana. Learning many related tasks at the same time with backpropagation. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems 7*, pages 657–664, San Mateo, CA, 1995. Morgan Kaufmann.

[116] T. Cavalier-Smith. Introduction: The evolutionary significance of genome size. In T. Cavalier-Smith, editor, *The Evolution of Genome Size*, pages 1–36. John Wiley & Sons, Chichester, UK, 1985.

[117] T. Cavalier-Smith. The origin of cells: A symbiosis between genes, catalysts, and membranes. *Cold Spring Harbor Symp. Quant. Biol.*, 52:805–824, 1987.

[118] J. M. Chandonia and M. Karplus. New methods for accurate prediction of protein secondary structure. *Proteins*, 35:293–306, 1999.

[119] E. Chargaff. Structure and function of nucleic acids as cell constituents. *Fed. Proc.*, 10:654-659, 1951.

[120] E. Chargaff. How genetics got a chemical education. *Ann. N. Y. Acad. Sci.*, 325:345-360, 1979.

[121] E. Charniak. Bayesian networks without tears. *AI Mag.*, 12:50-63, 1991.

[122] P. Cheeseman. An inquiry into computer understanding. *Comput. Intell.*, 4:57-142, 1988. With discussion.

[123] R. O. Chen, R. Felciano, and R. B. Altman. RIBOWEB: linking structural computations to a knowledge base of published experimental data. *ISMB*, 5:84-87, 1997.

[124] Y. Cheng and G. M. Church. Biclustering of expression data. In *Proceedings of the 2000 Conference on Intelligent Systems for Molecular Biology (ISMB00), La Jolla, CA*, pages 93-103. AAAI Press, Menlo Park, CA, 2000.

[125] G. I. Chipens, Y. U. I. Balodis, and L. E. Gnilomedova. Polarity and hydropathic properties of natural amino acids. *Ukrain. Biokhim. Zh.*, 63:20-29, 1991.

[126] Sung-Bae Cho and Jin H. Kim. An HMM/MLP architecture for sequence recognition. *Neural Comp.*, 7:358-369, 1995.

[127] C. Chotia. One thousand families for the molecular biologist. *Nature*, 357:543-544, 1992.

[128] P. Y. Chou and G. D. Fasman. Empirical predictions of protein conformations. *Ann. Rev. Biochem.*, 47:251-276, 1978.

[129] P.Y. Chou and G.D. Fasman. Prediction of the secondary structure of proteins from their amino acid sequence. *Adv. Enzymol. Relat. Areas Mol. Biol.*, 47:45-148, 1978.

[130] G. A. Churchill. Stochastic models for heterogeneous DNA sequences. *Bull. Mathem. Biol.*, 51:79-94, 1989.

[131] M. G. Claros, S. Brunak, and G. von Heijne. Prediction of n-terminal protein sorting signals. *Curr. Opin. Struct. Biol.*, 7:394-398, 1997.

[132] J-M. Claverie. What if there are only 30,000 human genes. *Science*, 291:1255-1257, 2001.

[133] N. Colloc'h and F. E. Cohen. Beta-breakers: An aperiodic secondary structure. *J. Mol. Biol.*, 221:603-613, 1991.

[134] Int. Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409:860-921, 2001.

[135] G. F. Cooper. The computational complexity of probabilistic inference using Bayesian belief networks. *Art. Intell.*, 42:393-405, 1990.

[136] J. L. Cornette, K. B. Cease, H. Margalit, J. L. Spouge, J. A. Berzofsky, and C. DeLisi. Hydrophobicity scales and computational techniques for detecting amphiphatic structures in proteins. *J. Mol. Biol.*, 195:659-685, 1987.

[137] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley, New York, 1991.

[138] R. T. Cox. Probability, frequency and reasonable expectation. *Am. J. Phys.*, 14:1-13, 1964.

[139] I. P. Crawford, T. Niermann, and K. Kirschner. Prediction of secondary structure by evolutionary comparison: application to the alpha subunit of tryptophan synthase. *Proteins*, 2:118-129, 1987.

[140] F. H. C. Crick. The origin of the genetic code. *J. Mol. Biol.*, 38:367-379, 1968.

[141] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines.* Cambridge University Press, Cambridge, UK, 2000.

[142] L. Croft, S. Schandorff, F. Clark, K. Burrage, P. Arctander, and J. S. Mattick. ISIS, the intron information system, reveals the high frequency of alternative splicing in the human genome. *Nature Genet.*, 24:340-341, 2000.

[143] S. Dalal, S. Balasubramanian, and L. Regan. Protein alchemy: Changing beta-sheet into alpha-helix. *Nat. Struct. Biol.*, 4:548-552, 1997.

[144] S. Das, L. Yu, C. Gaitatzes, R. Rogers, J. Freeman, J. Bienkowska, R. M. Adams, T. F. Smith, and J. Lindelien. Biology's new Rosetta Stone. *Nature*, 385:29-30, 1997.

[145] A. P. Dawid. Applications of a general propagation algorithm for probabilistic expert systems. *Stat. Comp.*, 2:25-36, 1992.

[146] P. Dayan, G. E. Hinton, R. M. Neal, and R. S. Zemel. The Helmholtz machine. *Neural Comp.*, 7:889-904, 1995.

[147] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Statis. Soc.*, B39:1-22, 1977.

[148] J. L. DeRisi, V. R. Iyer, and P. O. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 278:680-686, 1997.

[149] D. Devos and A. Valencia. Practical limits of function prediction. *Proteins*, 41:98-107, 2000.

[150] P. Diaconis and D. Stroock. Geometric bounds for eigenvalues of Markov chains. *Ann. Appl. Prob.*, 1:36-61, 1991.

[151] K. A. Dill, S. Bromberg, K. Yue, K. M. Fiebig, D. P. Yee, P. D. Thomas, and H. S. Chan. Principles of protein folding—a perspective from simple exact models. *Protein Sci.*, 4:561-602, 1995.

[152] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth. Hybrid Monte Carlo. *Phys. Lett. B*, 195:216-222, 1987.

[153] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis.* John Wiley and Sons, 1973.

[154] R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids.* Cambridge University Press, Cambridge, 1998.

[155] S. R. Eddy. Hidden Markov models. *Curr. Opin. Struct. Biol.*, 6:361-365, 1996.

[156] S. R. Eddy and R. Durbin. RNA sequence analysis using covariance models. *Nucl. Acids Res.*, 22:2079-2088, 1994.

[157] S. R. Eddy, G. Mitchinson, and R. Durbin. Maximum discrimination hidden Markov models of sequence consensus. *J. Comp. Biol.*, 2:9-23, 1995.

[158] H. Ehrig, M. Korff, and M. Lowe. Tutorial introduction to the algebraic approach of graph grammars based on double and single pushouts. In H. Ehrig, H. J. Kreowski, and G. Rozenberg, editors, *Lecture Notes in Computer Science*, volume

Kreowski, and G. Rozenberg, editors, *Lecture Notes in Computer Science*, volume 532, pages 24–37. Springer-Verlag, 1991.

[159] M. Eigen. The hypercycle. A principle of natural self-organization. Part A: Emergence of the hypercycle. *Naturwissenschaften*, 64:541–565, 1977.

[160] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci USA*, 95:14863–14868, 1998.

[161] D. Eisenberg. Into the black night. *Nat. Struct. Biol.*, 4:95-97, 1997.

[162] D. Eisenberg, E. M. Marcotte, and I. Xenarios T. O. Yeates. Protein function in the post-genomic era. *Nature*, 405:823–826, 2000.

[163] G. Elgar, R. Sandford, S. Aparicio, A. Macrae, B. Venkatesh, and S. Brenner. Small is beautiful: Comparative genomics with the pufferfish (*Fugu rubripes*). *Trends Genet.*, 12:145–150, 1996.

[164] J. Engelbrecht, S. Knudsen, and S. Brunak. G/C rich tract in the 5' end of human introns. *J. Mol. Biol.*, 227:108–113, 1992.

[165] J. Engelfriet and G. Rozenberg. Graph grammars based on node rewriting: An introduction to NLC graph grammars. In H. Ehrig, H. J. Kreowski, and G. Rozenberg, editors, *Lecture Notes in Computer Science*, volume 532, pages 12–23. Springer-Verlag, 1991.

[166] D. M. Engelman, T. A. Steitz, and A. Goldman. Identifying nonpolar transbilayer helices in amino acid sequences of membrane proteins. *Ann. Rev. Biophys. Biophys. Chem.*, 15:321–353, 1986.

[167] A. J. Enright, I. Iliopoulos, N. C. Kyrpides, and C. A. Ouzounis. Protein interaction maps for complete genomes based on gene fusion events. *Nature*, 402:86–90, 1999.

[168] C. J. Epstein. Role of the amino-acid "code" and of selection for conformation in the evolution of proteins. *Nature*, 210:25–28, 1966.

[169] D. T. Ross et al. Systematic variation in gene expression patterns in human cancer cell lines. *Nat. Genet.*, 24:227–235, 2000.

[170] J. C. Venter et al. The sequence of the human genome. *Science*, 291:1304–1351, 2001.

[171] U. Scherf et al. A gene expression database for the molecular pharmacology of cancer. *Nat. Genet.*, 24:236–244, 2000.

[172] B. S. Everitt. *An Introduction to Latent Variable Models*. Chapman and Hall, London, 1984.

[173] B. S. Everitt and D. J. Hand. *Finite Mixture Distributions*. Chapman and Hall, London and New York, 1981.

[174] P. Fariselli and R. Casadio. Prediction of the number of residue contacts in proteins. *ISMB*, 8:146–151, 19.

[175] B. A. Fedorov. Long-range order in globular proteins. *FEBS Lett.*, 62:139–141, 1976.

[176] W. Feller. *An Introduction to Probability Theory and its Applications*, volume 1. Jolm Wiley & Sons, New York, 3rd edition, 1968.

[177] W. Feller. *An Introduction to Probability Theory and its Applications*, volume 2. John Wiley & Sons, New York, 2nd edition, 1971.

[178] J. Felsenstein. Evolutionary trees from DNA sequences: A maximum likelihood approach. *J. Mol. Evol.*, 17:368–376, 1981.

[179] E. A. Ferran, B. Pflugfelder, and P. Ferrara. Self-organized neural maps of human protein sequences. *Protein Sci.*, 3:507–521, 1994.

[180] J. A. Fill. Eigenvalue bounds on convergence to stationarity for nonreversible Markov chains with an application to an exclusion process. *Ann. Appl. Prob.*, 1:62–87, 1991.

[181] W. M. Fitch. Towards defining the course of evolution: Minimum change for a specific tree topology. *Syst. Zool.*, 20:406–416, 1971.

[182] W. M. Fitch and E. Margoliash. Construction of phylogenetic trees. *Science*, 155:279–284, 1967.

[183] R. D. Fleischmann, M. D. Adams, O. White, R. A. Clayton, E. F. Kirkness, A. R. Kerlavage, C. J. Bult, J. F. Tomb, B. A. Dougherty, and J. M. Merrick. Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. *Science*, 269:496–512, 1995.

[184] M. L. Forcada and R. C. Carrasco. Learning the initial state of a second-order recurrent neural network during regular-language inference. *Neural Comp.*, 7:923–930, 1995.

[185] D. R. Forsdyke. Relative roles of primary sequence and (G+C)% in determining the hierarchy of frequencies of complementary trinucleotide pairs in dnas of different species. *J. Mol. Evol.*, 41:573–581, 1995.

[186] G. E. Fox and C. R. Woese. The architecture of 5S rRNA and its relation to function. *J. Mol. Evol.*, 6:61–76, 1975.

[187] V. Di Francesco, J. Garnier, and P. J. Munson. Protein topology recognition from secondary structure sequences—Applications of the hidden Markov models to the alpha class proteins. *J. Mol. Biol.*, 267:446–463, 1997.

[188] P. Frasconi, M. Gori, and A. Sperduti. A general framework for adaptive processing of data structures. *IEEE Trans. Neural Networks*, 9:768–786, 1998.

[189] C. M. Fraser, J. D. Gocayne, O. White, M. D. Adams, R. A. Clayton, R. D. Fleischmann, C. J. Bult, A. R. Kerlavage, G. Sutton, and J. M. Kelley. The minimal gene complement of *Mycoplasma genitalium*. *Science*, 270:397–403, 1995.

[190] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using Bayesian networks to analyze expression data. *J. Comp. Biol.*, 7:601–620, 2000.

[191] A. Frigessi, P. Di Stefano, C. R. Hwang, and S. J. Sheu. Convergence rate of the Gibbs sampler, the Metropolis algorithm and other single-site updating dynamics. *J. R. Stat. Soc.*, 55:205–219, 1993.

[192] D. Frishman and P. Argos. Knowledge-based secondary structure assignment. *Proteins*, 23:566–579, 1995.

[193] Y. Fujiwara, M. Asogawa, and A. Konagaya. Stochastic motif extraction using hidden Markov models. In *Proceedings of Second International Conference on Intelligent Systems for Molecular Biology*, pages 138–146, Menlo Park, CA, 1994. AAAI/MIT Press.

[194] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16:906–914, 2000.

[195] G. Gamow. Possible relation between deoxyribonucleic acid and protein structures. *Nature*, 173:318, 1954.

[196] J. Garnier, D. J. Osguthorpe, and B. Robson. Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. *J. Mol. Biol.*, 120:97–120, 1978.

[197] R. A. Garrett. Genomes: *Methanococcus jannaschii* and the golden fleece. *Curr. Biol.*, 6:1376–1377, 1996.

[198] A. Gelman and T. P. Speed. Characterizing a joint probability distribution by conditionals. *J. R. Statis. Soc. B*, 55:185–188, 1993.

[199] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Machine Intell.*, 6:721–741, 1984.

[200] D. Gerhold and C. T. Caskey. It's the genes! EST access to human genome content. *Bioessays*, 18:973–981, 1996.

[201] M. Gerstein, E. Sonnhammer, and C. Chotia. Volume changes in protein evolution. *J. Mol. Biol.*, 236:1067–1078, 1994.

[202] C. J. Geyer. Practical Markov chain Monte Carlo. *Statis. Sci.*, 7:473–511, 1992.

[203] Z. Ghahramani. Learning dynamic Bayesian networks. *Adap. Proc. Seq. Data Struct.*, 1387:168–197, 1998.

[204] Z. Ghahramani. Learning dynamic Bayesian networks. In M. Gori and C. L. Giles, editors, *Adaptive Processing of Temporal Information. Lecture Notes in Artifical Intelligence*. Springer Verlag, Heidelberg, 1998.

[205] Z. Ghahramani and M. I. Jordan. Factorial hidden Markov models. *Machine Learning*, 1997.

[206] M. Gibbs and D. J. C. MacKay. Efficient implementation of Gaussian processes. Technical report, Cavendish Laboratory, Cambridge, UK, 1997.

[207] L. M. Gierasch. Signal sequences. *Biochemistry*, 28:923–930, 1989.

[208] C. L. Giles, C. B. Miller, D. Chen, H. H. Chen, G. Z. Sun, and Y. C. Lee. Learning and extracting finite state automata with second-order recurrent neural networks. *Neural Comp.*, 4:393–405, 1992.

[209] W. R. Gilks, D. G. Clayton, D. J. Spiegelhalter, N. G. Best, A. J. McNeil, L. D. Sharples, and A. J. Kirby. Modelling complexity: Applications of Gibbs sampling in medicine. *J. R. Statis. Soc.*, 55:39–52, 1993.

[210] W. R. Gilks, A. Thomas, and D. J. Spiegelhalter. A language and program for complex Bayesian modelling. *The Statistician*, 43:69–78, 1994.

[211] P. Gill, P. L. Ivanov, C. Kimpton, R. Piercy, N. Benson, G. Tully, I. Evett, E. Hagelberg, and K. Sullivan. Identification of the remains of the Romanov family by DNA analysis. *Nat. Genet.*, 6:130–135, 1994.

[212] P. Gill, C. Kimpton, R. Aliston-Greiner, K. Sullivan, M. Stoneking, T. Melton, J. Nott, S. Barritt, R. Roby, and M. Holland. Establishing the identity of Anna Anderson Manahan. *Nat. Genet.*, 9:9–10, 1995.

[213] V. Giudicelli and M.-P. Lefranc. Ontology for Immunogenetics: IMGT-ONTOLOGY. *Bioinformatics*, 12:1047–1054, 1999.

[214] U. Gobel, C. Sander, R. Schneider, and A. Valencia. Correlated mutations and residue contacts in proteins. *Proteins*, 18:309–317, 1994.

[215] A. Goffeau. Life with 6000 genes. *Science*, 274:546, 1996.

[216] A. L. Goldberg and R. E. Wittes. Genetic code: Aspects of organization. *Science*, 153:420–424, 1966.

[217] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.

[218] D. S. Goodsell and R. E. Dickerson. Bending and curvature calculations in B-DNA. *Nucl. Acids Res.*, 22:5497–5503, 1994.

[219] J. Gorodkin, L. J. Heyer, S. Brunak, and G. D. Stormo. Displaying the information contents of structural RNA alignments: the structure logos. *CABIOS*, 13:583–586, 1997.

[220] J. Gorodkin, L. J. Heyer, and G. D. Stormo. Finding common sequence and structure motifs in a set of RNA sequences. In T. Gaasterland, P. Karp, K. Karplus, C. Ouzounis, C. Sander, and A. Valencia, editors, *Proceedings of the Fifth International Conference on Intellelligent Systems for Molecular Biology*, pages 120–123, Menlo Park, California, 1997. AAAI/MIT Press.

[221] J. Gorodkin, L. J. Heyer, and G. D. Stormo. Finding the most significant common sequence and structure motifs in a set of RNA sequences. *Nucl. Acids Res.*, 25:3724–3732, 1997.

[222] M. Gouy, P. Marliere, C. Papanicolaou, and J. Ninio. Prediction of secondary structures of nucleic acids: Algorithmic and physical aspects. *Biochimie*, 67:523–531, 1985.

[223] C. W. J. Granger. Combining forecasts—twenty years later. *J. Forecasting*, 8:167–173, 1989.

[224] P. Green, D. Lipman, L. Hillier, R. Waterson, D. States, and J. M. Claverie. Ancient conserved regions in new gene sequences and the protein databases. *Science*, 259:1711–1716, 1993.

[225] P. C. Gregory and T. J. Loredo. A new method for the detection of a periodic signal of unknown shape and period. *Astrophys. J.*, 398:146–168, 1992.

[226] M. Gribskov, A. D. McLachlan, and D. Eisenberg. Profile analysis: Detection of distantly related proteins. *Proc. Natl. Acad. Sci. USA*, 84:4355–4358, 1987.

[227] T. Gudermann, T. Schoneberg, and G. Schnltz. Functional and structural complexity of signal transduction via g-protein-coupled receptors. *Annu. Rev. Neurosci.*, 20:399–427, 1997.

[228] S. F. Gull. Bayesian inductive inference and maximum entropy. In G. J. Erickson and C. R. Smith, editors, *Maximum entropy and Bayesian methods in science and engineering*, pages 53–74. Kluwer, Dordrecht, 1988.

[229] S.F. Gull. Developments in maximum entropy data analysis. In J. Skilling, editor, *Maximum entropy and Bayesian methods*, pages 53–71. Kluwer, Dordrecht, 1989.

[230] B. Hajeck. Cooling schedules for optimal annealing. *Math. of Operation Res.*, 13:311-329, 1988.

[231] S. Hampson, P. Baldi, D. Kibler, and S. Sandmeyer. Analysis of yeast's ORFs upstream regions by parallel processing, microarrays, and computational methods. In *Proceedings of the 2000 Conference on Intelligent Systems for Molecular Biology (ISMB00), La Jolla, CA*, pages 190-201. AAAI Press, Menlo Park, CA, 2000.

[232] S. Hampson, D. Kibler, and P. Baldi. Distribution patterns of locally overrepresented *k*-mers in non-coding yeast DNA. 2001. Submitted.

[233] S. Handley. Classifying nucleic acid sub-sequences as introns or exons using genetic programming. In C. Rawlings, D. Clark, R. Altman, L. Hunter, T. Lengauer, and S. Wodak, editors, *Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology*, pages 162-169. AAAI Press, Menlo Park, CA, 1995.

[234] J. Hanke, D. Brett, I. Zastrow, A. Aydin, S. Delbruck, G. Lehmann, F. Luft, J. Reich, and P. Bork. Alternative splicing of human genes: more the rule than the exception? *Trends Genet.*, 15:389-390, 1999.

[235] J. E. Hansen, O. Lund, J. Engelbrecht, H. Bohr, J. O. Nielsen, J. E.-S. Hansen, and S. Brunak. Prediction of O-glycosylation of mammalian proteins: Specificity patterns of UDP-GalNAc:polypeptide n-acetylgalactosaminyltransferase. *J. Biochem. Biol.*, 307:801-813, 1995.

[236] J. E. Hansen, O. Lund, N. Tolstrup, A. A. Gooley, K. L. Williams, and S. Brunak. NetOglyc: prediction of mucin type O-glycosylation sites based on sequence context and surface accessibility. *Glycocon. J.*, 15:115-130, 1998.

[237] L. Hansen and P. Salamon. Neural network ensembles. *IEEE Trans. Pattern Anal. and Machine Intell.*, 12:993-1001, 1990.

[238] J. C. Harsanyi. *Rational behaviour and bargaining equilibrium in games and social situations.* Cambridge University Press, Cambridge, UK, 1977.

[239] L. H. Hartwell, J. J. Hopfield, S. Leibler, and A. W. Murray. From molecular to modular cell biology. *Nature*, 402, Supp.:C47-C52, 1999.

[240] M. Hasegawa and T. Miyata. On the antisymmetry of the amino acid code table. *Orig. Life*, 10:265-270, 1980.

[241] M. A. El Hassan and C. R. Calladine. Propeller-twisting of base-pairs and the conformational mobility of dinucleotide steps in DNA. *J. Mol. Biol.*, 259:95-103, 1996.

[242] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57:97-109, 1970.

[243] J. Hasty, J. Pradines, M. Dolnik, and J. J. Collins. Noise-based switches and amplifiers for gene expression. *Proc. Natl. Acad. Sci. USA*, 97:2075-2080, 2000.

[244] S. Hayward and J. F. Collins. Limits on α-helix prediction with neural network models. *Proteins*, 14:372-381, 1992.

[245] S. M. Hebsgaard, P. G. Korning, N. Tolstrup, J. Engelbrecht, P. Rouzé, and S. Brunak. Splice site prediction in *Arabidopsis thaliana* pre-mRNA by combining local and global sequence information. *Nucl. Acids Res.*, 24:3439-3452, 1996.

[246] D. Heckerman. Bayesian networks for data mining. *Data Mining and Knowl.*

Discov., 1:79–119, 1997.

[247] J. Hein. Unified approach to alignment and phylogenies. *Meth. Enzymol.*, 183:626–645, 1990.

[248] R. Henderson, J. M. Baldwin, T. A. Ceska, F. Zemlin, E. Beckmann, and K. H. Downing. Model for the structure of bacteriorhodopsin based on high-resolution electron cryo-microscopy. *J. Mol. Biol.*, 213:899–929, 1990.

[249] S. Henikoff and J. Henikoff. Position-based sequence weights. *J. Mol. Biol.*, 243:574–578, 1994.

[250] S. Henikoff and J. G. Henikoff. Protein family classification based on searching a database of blocks. *Genomics*, 19:97–107, 1994.

[251] B. Hermann and S. Hummel, editors. *Ancient DNA*. Springer-Verlag, New York, 1994.

[252] J. Hertz, A. Krogh, and R.G. Palmer. *Introduction to the theory of neural computation*. Addison-Wesley, Redwood City, CA, 1991.

[253] L. J. Heyer, S. Kruglyak, and S. Yooseph. Exploring expression data: identification and analysis of co-expressed genes. *Genome Res.*, 9:1106–1115, 1999.

[254] R. Hinegardner. Evolution of cellular DNA content in teleost fishes. *Am. Nat.*, 102:517–523, 1968.

[255] G. E. Hinton, P. Dayan, B. J. Frey, and R. M. Neal. The wake-sleep algorithm for unsupervised neural networks. *Science*, 268:1158–1161, 1995.

[256] H. Le Hir, M. J. Moore, and L. E. Maquat. Pre-mrna splicing alters mRNP composition: evidence for stable association of proteins at exon-exon junctions. *Genes Dev.*, 14:1098–1108, 2000.

[257] R. Hirata, Y. Ohsumk, A. Nakano, H. Kawasaki, K. Suzuki, and Y. Anraku. Molecular structure of a gene, vma1, encoding the catalytic subunit of H(+)- translocating adenosine triphosphatase from vacuolar membranes of *Saccharomyces cerevisiae*. *J. Biol. Chem.*, 265:6726–6733, 1990.

[258] W. S. Hlavacek and M. S. Savageau. Completely uncoupled and perfectly coupled gene expression in repressible systems. *J. Mol. Biol.*, 266:538–558, 1997.

[259] U. Hobohm, M. Scharf, R. Schneider, and C. Sander. Selection of representative protein data sets. *Protein Sci.*, 1:409–417, 1992.

[260] I. L. Hofacker, W. Fontana, P. F. Stadler, S. Bonherffer, M. Tacker, and P. Schuster. Fast folding and comparison of RNA secondary structures. *Monatshefte f. Chemie*, 125:167–188, 1994.

[261] J. H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT Press, Cambridge, MA, 1992.

[262] L. H. Holley and M. Karplus. Protein secondary structure prediction with a neural network. *Proc. Nat. Acad. Sci. USA*, 86:152–156, 1989.

[263] F. C. P. Holstege, E. G. Jennings, J. J. Wyrick, T. I. Lee, C. J. Hengartner, M. R. Green, T. R. Golub, E. S. Lander, and R. A. Young. Dissecting the regulatory circuitry of a eukaryotic genome. *Cell*, 95:717–728, 1998.

[264] K. Hornik, M. Stinchcombe, and H. White. Universal approximation of an unknown function and its derivatives using multilayer feedforward networks. *Neural Networks*, 3:551–560, 1990.

[265] K. Hornik, M. Stinchcombe, H. White, and P. Auer. Degree of approximation results for feedforward networks approximating unknown mappings and their derivatives. *Neural Comp.*, 6:1262–1275, 1994.

[266] Z. Huang, S. B. Prusiner, and F. E. Cohen. Scrapie prions: A three-dimensional model of an infectious fragment. *Folding & Design*, 1:13–19, 1996.

[267] Z. Huang, S. B. Prusiner, and F. E. Cohen. Structures of prion proteins and conformational models for prion diseases. *Curr. Top. Microbiol. Immunol.*, 207:49–67, 1996.

[268] T. J. Hubbard and J. Park. Fold recognition and ab initio structure predictions using hidden Markov models and beta-strand pair potentials. *Proteins*, 25:398–402, 1995.

[269] J. P. Huelsenbeck and B. Rannala. Phylogenetic methods come of age: Testing hypotheses in an evoutionary context. *Science*, 276:227–232, 1997.

[270] J. D. Hughes, P. W. Estep, S. Tavazole, and G. M. Church. Computational identification of *cis*-regulatory elements associated with groups of functionally related genes in *saccharomyces cerevisiae*. *J. Mol. Biol.*, 296:1205–1214, 2000.

[271] T. R. Hughes, M. J. Marton, A. R. Jones, C. J. Roberts, R. Stoughton, C. D. Armour, H. A. Bennett, E. Coffey, H. Dai, Y. D. He, K. J. Kidd, A. M. King, M. R. Meyer, D. Slade, P. Y. Lum, S. B. Stepaniants, D. D. Shoemaker, D. Gachotte, K. Chakraburtty, J. Simon, M. Bard, and S. H. Friend. Functional discovery via a compendium of expression profiles. *Cell*, 102:109–126, 2000.

[272] V. Isham. An introduction to spatial point processes and Markov random fields. *Internat. Statist. Rev.*, 49:21–43, 1981.

[273] O. C. Ivanov and B. Förtsch. Universal regularities in protein primary structure: preference in bonding and periodicity. *Orig. Life Evol. Biosph.*, 17:35–49, 1986.

[274] P. L. Ivanov, M. J. Wadhams, R. K. Roby, M. M. Holland, V. W. Weedn, and T. J. Parsons. Mitochondrial DNA sequence heteroplasmy in the grand duke of Russia Georgij Romanov establishes the authenticity of the remains of Tsar Nicholas II. *Nat. Genet.*, 12:417–420, 1996.

[275] T. S. Jaakkola, M. Diekhans, and D. Haussler. Using the Fisher kernel method to detect remore protein homologies. In T. Lengauer, R. Schneider, P. Bork, D. Brutlag, J. Glasgow, H. W. Mewes, and R. Zimmer, editors, *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology (ISMB99)*, pages 149–155. AAAI Press, Menlo Park, CA, 1999.

[276] T. S. Jaakkola and I. Jordan. Recursive algorithms for approximating probabilities in graphical models. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 487–493. MIT Press, Cambridge, MA, 1997.

[277] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural Comp.*, 3:79–87, 1991.

[278] B. D. James, G. J. Olsen, and N. R. Pace. Phylogenetic comparative analysis of RNA secondary structure. *Meth. Enzymol.*, 180:227–239, 1989.

[279] P. G. Jansen. *Exploring the exon universe using neural networks*. PhD thesis, The Technical University of Denmark, 1993.

[280] E. T. Jaynes. Information theory and statistical mechanics. *Phys. Rev.*, 106:620–

630, 1957.

[281] E. T. Jaynes. Information theory and statistical mechanics. II. *Phys. Rev.*, 108:171–190, 1957.

[282] E. T. Jaynes. Prior probabilities. *IEEE Trans. Systems Sci. Cybernet.*, 4:227–241, 1968.

[283] E. T. Jaynes. Bayesian methods: General background. In J. H. Justice, editor, *Maximum entropy and Bayesian methods in statistics*, pages 1–25. Cambridge University Press, Cambridge, 1986.

[284] E. T. Jaynes. Probability theory: The logic of science. Unpublished., 1994.

[285] W. H. Jeffreys and J. O. Berger. Ockham's razor and Bayesian analysis. *Am. Sci.*, 80:64–72, 1992.

[286] F. V. Jensen. *An Introduction to Bayesian Networks*. Springer Verlag, New York, 1996.

[287] F. V. Jensen, S. L. Lauritzen, and K. G. Olesen. Bayesian updating in causal probabilistic networks by local computations. *Comput. Statist. Quart.*, 4:269–282, 1990.

[288] L. J. Jensen, R. Gupta, N. Blom, D. Devos, J. Tamames, C. Kesmir, H. Nielsen, C. Workman, C. A. Andersen, K. Rapacki, H.H. Stærfelt, A. Krogh, S. Knudsen, A. Valencia, and S. Brunak. Using posttranslational modifications to predict orphan protein function for the human genome. *Submitted*, 2001.

[289] H. Jeong, B. Tomber, R. Albert, Z.N. Oltvai, and A.-L. Barabasi. The large-scale organization of metabolic networks. *Nature*, 407:651–654, 2000. in press.

[290] D. T. Jones. Protein secondary structure prediction based on position-specific scoring matrices. *J. Mol. Biol.*, 292:195–202, 1999.

[291] D. T. Jones, C. M. Moody, J. Uppenbrink, J. H. Viles, P. M. Doyle, C. J. Harris, L. H. Pearl, P. J. Sadler, and J. M. Thornton. Towards meeting the Paracelsus challenge: The design, synthesis, and characterization of paracelsin-43, an alpha-helical protein with over 50% sequence identity to an all-beta protein. *Proteins*, 24:502–513, 1996.

[292] M. I. Jordan. *Learning in Graphical Models*. MIT Press, Cambridge, MA, 1999.

[293] M. I. Jordan, Z. Ghahramani, and L. K. Saul. Hidden Markov decision trees. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 501–507. MIT Press, Cambridge, MA, 1997.

[294] T. H. Jukes. Possibilities for the evolution of the genetic code from a preceding form. *Nature*, 246:22–26, 1973.

[295] T. H. Jukes and C. R. Cantor. Evolution of protein molecules. In *Mammalian Protein Metabolism*, pages 21–132. Academic Press, New York, 1969.

[296] B. Jungnickel, T.A. Rapoport, and E. Hartmann. Protein translocation: Common themes from bacteria to man. *FEBS Lett.*, 346:73–77, 1994.

[297] W. Kabsch and C. Sander. Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22:2577–2637, 1983.

[298] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement learning: A survey. *J. Art. Intell. Res.*, 4:237–285, 1996.

[299] P. Kahn. From genome to proteome: Looking at a cell's proteins. *Science*, 270:369–370, 1995.

[300] D. Kaiser and R. Losick. How and why bacteria talk to each other. *Cell*, 73:873–885, 1993.

[301] P. M. Kane, C. T. Yamashiro, D. F. Wolczyk, N. Neff, M. Goebl, and T. H. Stevens. Protein splicing converts the yeast tfp1 gene product to the 69-kd subunit of the vacuolar h(+)-adenosine triphosphatase. *Science*, 250:651–657, 1990.

[302] N. Kaplan and C. H. Langley. A new estimate of sequence divergence of mitochondrial DNA using restriction endonuclease mappings. *J. Mol. Evol.*, 13:295–304, 1979.

[303] J. D. Karkas, R. Rudner, and E. Chargaff. Separation of *B. subtilis* DNA into complementary strands, II. Template functions and composition as determined by transcription with RNA polymerase. *Proc. Natl. Acad. Sci. USA*, 60:915–920, 1968.

[304] S. Karlin, B. E. Blaisdell, and P. Bucher. Quantile distributions of amino acid usage in protein classes. *Prot. Eng.*, 5:729–738, 1992.

[305] S. Karlin and J. Mrazek. What drives codon choices in human genes. *J. Mol. Biol.*, 262:459–472, 1996.

[306] S. Karlin, F. Ost, and B. E. Blaisdell. Patterns in DNA and amino acid sequences and their statistical significance. In M.S. Waterman, editor, *Mathematical methods for DNA sequences*, pages 133–157, Boca Raton, Fla., 1989. CRC Press.

[307] P. Karp, M. Riley, S. Paley, A. Pellegrini-Toole, and M. Krummenacker. EcoCyc: Electronic encyclopedia of *e. coli* genes and metabolism. *Nucl. Acids Res.*, 27:55–59, 1999.

[308] P. D. Karp. An ontology for biological function based on molecular interactions. *Bioinformatics*, 16:269–285, 2000.

[309] P. D. Karp, M. Krummenacker, S. Paley, and J. Wagg. Integrated pathway-genome databases and their role in drug discovery. *Trends Biotech.*, 17:275–281, 1999.

[310] S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theor. Biol.*, 22:437–467, 1969.

[311] S. A. Kauffman. The large scale structure and dynamics of gene control circuits: an ensemble approach. *J. Theor. Biol.*, 44:167–190, 1974.

[312] S. A. Kauffman. Requirements for evolvability in complex systems: orderly dynamics and frozen components. *Physica D*, 42:135–152, 1990.

[313] T. Kawabata and J. Doi. Improvement of protein secondary structure prediction using binary word encoding. *Proteins*, 27:36–46, 1997.

[314] M. J. Kearns and U. V. Vazirani. *An Introduction to Computational Learning Theory*. MIT Press, Cambridge, MA, 1994.

[315] W. J. Kent and A. M. Zahler. The Intronerator: exploring introns and alternative splicing in caenorhabditis elegans. *Nucl. Acids Res.*, 28:91–93, 2000.

[316] D. H. Kenyon and G. Steinman. *Biochemical Predestinations*. McGraw-Hill, New York, 1969.

[317] H. G. Khorana. Bacteriorhodopsin, a membrane protein that uses light to translocate protons. *J. Biol. Chem.*, 263:7439–7442, 1988.

[318] H. G. Khorana, G. E. Gerber, W. C. Herlihy, C. P. Gray, R. J. Anderegg, K. Nihei, and K. Biemann. Amino acid sequence of bacteriorhodopsin. *Proc. Natl. Acad. Sci.*, 76:5046–5050, 1979.

[319] J. L. King and T. H. Jukes. Non-Darwinian evolution. *Science*, 164:788–798, 1969.

[320] R. D. King and M. J. Sternberg. Identification and application of the concepts important for accurate and reliable protein secondary structure prediction. *Prot. Sci.*, 5:2298–2310, 1996.

[321] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220:671–680, 1983.

[322] T. M. Klingler and D. L. Brutlag. Discovering side-chain correlation in alpha-helices. In R. Altman, D. Brutlag, P. Karp, R. Lathrop, and D. Searls, editors, *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pages 236–243. AAAI Press, Menlo Park, CA, 1994.

[323] D. G. Kneller, F. E. Cohen, and R. Langridge. Improvements in protein secondary structure prediction by an enhanced neural network. *J. Mol. Biol.*, 214:171–182, 1990.

[324] P. Koehl and M. Levitt. A brighter future for protein structure prediction. *Nat. Struct. Biol.*, 6:108–111, 1999.

[325] L. F. Kolakowski. GCRDb: A G-protein-coupled receptor database. *Receptors Channels*, 2:1–7, 1994.

[326] A. K. Konopka. Sequences and codes: Fundamentals of biomolecular cryptology. In D. W. Smith, editor, *Biocomputing—Informatics and Genome Projects*, pages 119–174, San Diego, 1994. Academic Press.

[327] P. G. Korning, S. M. Hebsgaard, P. Rouze, and S. Brunak. Cleaning the GenBank *Arabidopsis thaliana* data set. *Nucl. Acids Res.*, 24:316–320, 1996.

[328] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection.* MIT Press, Cambridge MA, 1992.

[329] J. R. Koza. Evolution of a computer program for classifying protein segments as transmembrane domains using genetic programming. In R. Altman, D. Brutlag, P. Karp, R. Lathrop, and D. Searls, editors, *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pages 244–252. AAAI Press, Menlo Park, CA, 1994.

[330] J. R. Koza. *Genetic Programming II: Automatic Discovery of Reusable Programs.* MIT Press, Cambridge MA, 1994.

[331] A. Kreegipuu, N. Blom, S. Brunak, and J. Jarv. Statistical analysis of protein kinase specificity determinants. *FEBS Lett.*, 430:45–50, 1998.

[332] J. K. Kristensen. Analysis of cis alternatively spliced mammalian genes. Master Thesis, University of Copenhagen, 2000.

[333] A. Krogh. Two methods for improving performance of an HMM and their application for gene finding. In T. Gaasterland et al., editor, *Proceedings of the Fifth International Conference on Intelligent Systems for Molecular Biology*, pages 179–186. AAAI Press, Menlo Park, CA, 1997.

[334] A. Krogh, M. Brown, I. S. Mian, K. Sjölander, and D. Haussler. Hidden Markov models in computational biology: Applications to protein modeling. *J. Mol. Biol.*, 235:1501–1531, 1994.

[335] A. Krogh, B. Larsson, G. von Heijne, and E. L. Sonnhammer. Predicting transmembrane protein topology with a hidden markov model: application to complete genomes. *J. Mol. Biol.*, 305:567-580, 2001.

[336] A. Krogh, I. S. Mian, and D. Haussler. A hidden Markov model that finds genes in *E. coli* DNA. *Nucl. Acids Res.*, 22:4768-4778, 1994.

[337] A. Krogh and G. Mitchinson. Maximum entropy weighting of aligned sequences of proteins of DNA. In C. Rawlings, D. Clark, R. Altman, L. Hunter, T. Lengauer, and S. Wodak, editors, *Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology*, pages 215-221. AAAI Press, Menlo Park, CA, 1995.

[338] A. Krogh and S. K. Riis. Prediction of beta sheets in proteins. In M. C. Mozer S. Touretzky and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 917-923. MIT Press, Boston, MA, 1996.

[339] A. Krogh and P. Sollich. Statistical mechanics of ensemble learning. *Phys. Rev. E*, 55:811-825, 1997.

[340] A. Krogh and J. Vedelsby. Neural network ensembles, cross validation and active learning. In G. Tesauro, D. S. Touretzky, and T. K. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 231-238. MIT Press, Cambridge, MA, 1995.

[341] S. Kullback. *Information Theory and Statistics*. Dover Publications, New York, 1959.

[342] S. Kullback and R. A. Leibler. On information and sufficiency. *Ann. Math. Stat.*, 22:79, 1986.

[343] D. Kulp, D. Haussler, M. G. Reese, and F. H. Eeckman. A generalized hidden Markov model for the recognition of human genes in dna. *ISMB*, 4:134-142, 1996.

[344] A. Lapedes, C. Barnes, C. Burks, R. Farber, and K. Sirotkin. Application of neural networks and other machine learning algorithms to dna sequence analysis. In G. I. Bell and T. G. Marr, editors, *Computers in DNA. The Proceedings of the Interface Between Computation Science and Nucleic Acid Sequencing Workshop.*, volume VII, pages 157-182. Addison Wesley, Redwood City, CA, 1988.

[345] K. Lari and S. J. Young. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Lang.*, 4:35-36, 1990.

[346] N. Larsen, G. J. Olsen, B. L. Maidak, M. J. McCaughey, R. Overbeek, T. J. Macke, T. L. Marsh, and C. R. Woese. The ribosomal database project. *Nucl. Acids Res.*, 21:3021-3023, 1993.

[347] E. E. Lattman and G. D. Rose. Protein folding-what's the question? *Proc. Natl. Acad. Sci. USA*, 90:439-441, 1993.

[348] S. L. Lauritzen. *Graphical Models*. Oxford University Press, Oxford, 1996.

[349] S. L. Lauritzen, A. P. Dawid, B. N. Larsen, and H. G. Leimer. Independence properties of directed Markov fields. *Networks*, 20:491-505, 1990.

[350] S. L. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *J. R. Statis. Soc. B*, 50:157-224, 1988.

[351] C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. Neuwald, and J. C. Wootton. Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment. *Science*, 262:208-214, 1993.

[352] C. E. Lawrence and A. A. Reilly. An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. *Proteins*, 7:41-51, 1990.

[353] J. R. Lawton, F. A. Martinez, and C. Burks. Overview of the LiMB database. *Nucl. Acids Res.*, 17:5885-5899, 1989.

[354] C. Lee, R. G. Klopp, R. Weindruch, and T. A. Prolla. Gene expression profile of aging and its retardation by caloric restriction. *Science*, 285:1390-1393, 1999.

[355] M. T. Lee, F. C. Kuo, G. A. Whitmore, and J. Sklar. Importance of replication in microarray gene expression studies: statistical methods and evidence from repetitive cDNA hybridizations. *Proc. Natl. Acad. Sci. USA*, 97:9834-9839, 2000.

[356] N. Lehman and G. F. Joyce. Evolution in vitro of an RNA enzyme with altered metal dependence. *Nature*, 361:182-185, 1993.

[357] E. Levin and R. Pieraccini. Planar hidden Markov modeling: From speech to optical character recognition. In S. J. Hanson, J. D. Cowan, and C. Lee Giles, editors, *Advances in Neural Information Processing Systems*, volume 5, pages 731-738. Morgan Kaufmann, San Mateo, CA, 1993.

[358] J. Levin, S. Pascarella, P. Argos, and J. Garnier. Quantification of secondary structure prediction improvement using multiple alignments. *Prot. Eng.*, 6:849-854, 1993.

[359] S. E. Levinson, L. R. Rabiner, and M. M. Sondhi. An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition. *Bell Syst. Tech. J.*, 62:1035-1074, 1983.

[360] S. Lewis, M. Ashburner, and M. G. Reese. Annotating eukaryote genomes. *Curr. Opin. Struct. Biol.*, 10:349-354, 2000.

[361] F. Liang, I. Holt, G. Pertea, S. Karamycheva, S.L. Salzberg, and J. Quackenbush. Gene index analysis of the human genome estimates approximately 120, 000 genes. *Nat. Genetics*, 25:239-240, 200.

[362] V. I. Lim. Algorithms for prediction of α-helical and β-structural regions in globular proteins. *J. Mol. Biol.*, 88:873-894, 1974.

[363] S. Lin and A. D. Riggs. The general affinity of lac repressor for *E. coli* DNA: Implications for gene regulation in procaryotes and eucaryotes. *Cell*, 4:107-111, 1975.

[364] T. Lin, B. G. Horne, P. Tiño, and C. L. Giles. Learning long-term dependencies in NARX recurrent neural networks. *IEEE Trans. Neural Networks*, 7:1329-1338, 1996.

[365] H. Lodish, D. Baltimore, A. Berk, S. L. Zipursky, and P. Matsudairas. *Molecular cell biology*. Scientific American Books, New York, 3rd edition, 1995.

[366] A. D. Long, H. J. Mangalam, B. Y. Chan, L. Tolleri, G. W. Hatfield, and P. Baldi. Global gene expression profiling in *escherichia coli* K12: Improved statistical inference from DNA microarray data using analysis of variance and a Bayesian statistical framework. *J. Biol. Chem.*, 276:19937-19944, 2001.

[368] O. Lund, K. Frimand, J. Gorodkin, H. Bohr, J. Bohr, J. Hansen, and S. Brunak. Protein distance constraints predicted by neural networks and probability density functions. *Prot. Eng.*, 25:1241-1248, 1997.

[369] D. H. Ly, D. J. Lockhart, R. A. Lerner, and P. G. Schultz. Mitotic misregulation and human aging. *Science*, 287:2486-2492, 2000.

[370] M. J. MacGregor, T. P. Flores, and M. J. E. Sternberg. Prediction of beta-turns in proteins using neural networks. *Prot. Eng.*, 2:521-526, 1989.

[371] A. L. Mackay. Optimization of the genetic code. *Nature*, 216:159-160, 1967.

[372] D. J. C. MacKay. Bayesian interpolation. *Neural Comp.*, 4:415-447, 1992.

[373] D. J. C. MacKay. A practical Bayesian framework for back-propagation networks. *Neural Comp.*, 4:448-472, 1992.

[374] D. J. C. MacKay. Density networks and their application to protein modelling. In J. Skilling and S. Sibisi, editors, *Maximum Entropy and Bayesian Methods*, pages 259-268, Dordrecht, 1996. Kluwer.

[375] D. J. C. MacKay. Comparison of approximate methods for handling hyperparameters. *Neural Comp.*, 11:1035-1068, 1999.

[376] D. J. C. MacKay and L. C. Bauman Peto. A hierarchical Dirichlet language model. *Nat. Lang. Eng.*, 1:1-19, 1995.

[377] R. Maclin and J. Shavlik. Using knowledge-based neural networks to improve algorithms: Refining the Chou-Fasman algorithm for protein folding. *Machine Learning*, 11:195-215, 1993.

[378] E. M. Marcotte. Computational genetics: finding protein function by nonhomology methods. *Curr. Opin. Struct. Biol.*, 10:359-365, 2000.

[379] E. M. Marcotte, M. Pellegrini, H. L. Ng, D. L. Rice, T. O. Yeates, and D. Eisenberg. Detecting protein function and protein-protein interactions from genome sequences. *Science*, 285:751-753, 1999.

[380] E. M. Marcotte, M. Pellegrini, M. J. Thompson, T. O. Yeates, and D. Eisenberg. A combined algorithm for genome-wide prediction of protein function. *Nature*, 402:83-86, 1999.

[381] E. Marinari and G. Parisi. Simulated tempering: A new Monte Carlo scheme. *Europhys. Lett.*, 19:451-458, 1992.

[382] B. W. Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochim. Biophys. Acta*, 405:442-451, 1975.

[383] H. H. McAdams and A. Arkin. It's a noisy business! Genetic regulation at the nanomolar scale. *Trends Genet.*, 15:65-69, 1999.

[384] P. McCullagh and J. A. Nelder. *Generalized linear models*. Chapman and Hall, London, 1989.

[385] R. J. McEliece, D. J. C. MacKay, and J. F. Cheng. Turbo decoding as an instance of Pearl's belief propagation algorithm. *IEEE J. Sel. Areas Commun.*, 16:140-152, 1998.

[386] L. J. McGuffin, K. Bryson, and J. T. Jones. The PSIPRED protein structure prediction server. *Bioinformatics*, 16:404-405, 2000.

[387] X. L. Meng and D. B. Rubin. Recent extensions to the EM algorithm. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, editors, *Bayesian statistics*, volume 4, pages 307–320. Oxford University Press, Oxford, 1992.

[388] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equations of state calculations by fast computing machines. *J. Chem. Phys.*, 21:1087–1092, 1953.

[389] F. Miescher. Uber die chemische Zusammensetzung der Eiterzellen. In F. Hoppe-Seyler, editor, *Medicinisch-chemische Untersuchungen*, pages 441–460, Berlin, 1871. August Hirschwald.

[390] G. L. G. Miklos and G. M. Rubin. The role of the genome project in determining gene function: Insights from model organisms. *Cell*, 86:521–529, 1996.

[391] E. Mjolsness, D. H. Sharp, and J. Reinitz. A connectionist model of development. *J. Theor. Biol.*, 152:429–453, 1991.

[392] J. M. Modestino and J. Zhang. A Markov random field model-based approach to image interpretation. *IEEE Trans. Pattern Anal. Machine Intell.*, 14:606–615, 1992.

[393] R. N. Moll, M. A. Arbib, and A. J. Koufry. *An Introduction to Formal Language Theory*. Springer-Verlag, New York, 1988.

[394] J. C. Mullikin, S. E. Hunt, C. G. Cole, B. J. Mortimore, C. M. Rice, J. Burton, L. H. Matthews, R. Pavitt, R. W. Plumb, S. K. Sims, R. M. Ainscough, J. Attwood, J. M. Bailey, K. Barlow, R. M. Bruskiewich, P. N. Butcher, N. P. Carter, Y. Chen, C. M. Clee, P. C. Coggill, J. Davies, R. M. Davies, E. Dawson, M.D. Francis, A. A. Joy, R. G. Lamble, C. F. Langford, J. Macarthy, V. Mall, A. Moreland, E. K. Overton-Larty, M. T. Ross, L. C. Smith, C. A. Steward, J. E. Sulston, E. J. Tinsley, K. J. Turney, D. L. Willey, G. D. Wilson, A. A. McMurray, I. Dunham, J. Rogers, and D. R. Bentley. An SNP map of human chromosome 22. *Nature*, 407:516–520, 2000.

[395] R. M. Neal. Connectionist learning of belief networks. *Art. Intell.*, 56:71–113, 1992.

[396] R. M. Neal. Probabilistic inference using Markov chain Monte Carlo methods. Technical report. Department of Computer Science, University of Toronto, 1993.

[397] R. M. Neal. *Bayesian Learning for Neural Networks*. PhD thesis, Department of Computer Science, University of Toronto, 1995.

[398] R. M. Neal. *Bayesian Learning for Neural Networks*. Springer-Verlag, New York, 1996.

[399] R. M. Neal. Monte Carlo implementation of Gaussian process models for Bayesian regression and classification. Technical Report no. 9702. Department of Statistics, University of Toronto, 1997.

[400] R. M. Neal and G. E. Hinton. A new view of the EM algorithm that justifies incremental and other variants. Technical Report, Department of Computer Science, University of Toronto, Canada, 1993.

[401] S. B. Needleman and C. D Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, 48:443–453, 1970.

[402] E. J. Neer. G proteins: Critical control points for transmembrane signals. *Prot. Sci.*, 3:3-14, 1994.

[403] M. A. Newton, C. M. Kendziorski, C. S. Richmond, F. R. Blattner, and K. W. Tsui. On differential variability of expression ratios: improving statistical inference about gene expression changes from microarray data. *J. Comp. Biol.*, 8:37-52, 2001.

[404] H. Nielsen, J. Engelbrecht, S. Brunak, and G. von Heijne. Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Prot. Eng.*, 10:1-6, 1997.

[405] H. Nielsen, J. Engelbrecht, G. von Heijne, and S. Brunak. Defining a similarity threshold for a functional protein sequence pattern: The signal peptide cleavage site. *Proteins*, 24:316-320, 1996.

[406] H. Nielsen and A. Krogh. Prediction of signal peptides and signal anchors by a hidden Markov model. *ISMB*, 6:122-130, 1998.

[407] M. W. Nirenberg, O. W. Jones, P. Leder, B. F. C. Clark, W. S. Sly, and S. Pestka. On the coding of genetic information. *Cold Spring Harbor Symp. Quant. Biol.*, 28:549-557, 1963.

[408] R. Nowak. Entering the postgenome era. *Science*, 270:368-371, 1995.

[409] L. E. Orgel. A possible step in the origin of the genetic code. *Isr. J. Chem.*, 10:287-292, 1972.

[410] R. L. Ornstein, R. Rein, D. L. Breen, and R. D. MacElroy. An optimised potential function for the calculation of nucleic acid interaction energies. I. Base stacking. *Biopolymers*, 17:2341-2360, 1978.

[411] Y. A. Ovchinnikov, N. G. Abdulaev, M. Y. Feigina, A. V. Kiselev, and N. A. Lobanov. The structural basis of the functioning of bacteriorhodopsin: An overview. *FEBS Lett.*, 100:219-234, 1979.

[412] M. Pagel and R. A. Johnstone. Variation across species in the size of the nuclear genome supports the junk-DNA explanation for the c-value paradox. *Proc. R. Soc. Lond. (Biol.)*, 249:119-124, 1992.

[413] A. Pandey and M. Mann. Proteomics to study genes and genomes. *Nature*, 405:837-846, 2000.

[414] L. Pardo, J. A. Ballesteros, R. Osman, and H. Weinstein. On the use of the transmembrane domain of bacteriorhodopsin as a template for modeling the three-dimensional structure of guanine nucleotide-binding regulatory protein-coupled receptors. *Proc. Natl. Acad. Sci. USA*, 89:4009-4012, 1992.

[415] R. Parsons and M. E. Johnson. DNA sequence assembly and genetic programming—new results and puzzling insights. In C. Rawlings, D. Clark, R. Altman, L. Hunter, T. Lengauer, and S. Wodak, editors, *Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology*, pages 277-284. AAAI Press, Menlo Park, CA, 1995.

[416] J. Pearl. *Probabilistic Reasoning in Intelligent Systems.* Morgan Kaufmann, San Mateo, CA, 1988.

[417] W. R. Pearson. Rapid and sensitive sequence comparison with FASTP and FASTA.

Meth. Enzymol., 183:63–98, 1990.

[418] W. R. Pearson and D. J. Lipman. Improved tools for biological sequence comparison. *Proc. Nat. Acad. Sci. USA*, 85:2444–2448, 1988.

[419] W.R. Pearson. Effective protein sequence comparison. *Meth. Enzymol.*, 266:227–258, 1996.

[420] E. Pebay-Peyroula, G. Rummel, J. P. Rosenbusch, and E. M. Landau. X-ray structure of bacteriorhodopsin at 2.5 angstroms from microcrystals grown in lipidic cubic phases. *Science*, 277:1676–1681, 1997.

[421] A. G. Pedersen, P. F. Baldi, Y. Chauvin, and S. Brunak. DNA structure in human polymerase II promoters. *J. Mol. Biol.*, 281:663–673, 1998.

[422] A. G. Pedersen and H. Nielsen. Neural network prediction of translation initiation sites in eukaryotes: Perspectives for EST and genome analysis. In *Proceedings of the Fifth International Conference on Intelligent Systems for Molecular Biology*, pages 226–233, Menlo Park, CA., 1997. AAAI Press.

[423] M. Pellegrini, E. M. Marcotte, M. J. Thompson, D. Eisenberg, and T.O Yeates. Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. *Proc. Natl. Acad. Sci. U.S.A.*, 38:667–677, 1999.

[424] M. D. Perlwitz, C. Burks, and M. S. Waterman. Pattern analysis of the genetic code. *Advan. Appl. Math.*, 9:7–21, 1988.

[425] C. M. Perou, S. S. Jeffrey, M. van de Rijn, C. A. Rees, M. B. Eisen, D. T. Ross, A. Pergamenschikov, C. F. Williams, S. X. Zhu, J. C. Lee, D. Lashkari, D. Shalon, P. O. Brown, and D. Botstein. Distinctive gene expression patterns in human mammary epithelial cells and breast cancers. *Proc. Natl. Acad. Sci. USA*, 96:9212–9217, 1999.

[426] M. P. Perrone and L. N. Cooper. When networks disagree: ensemble method for neural networks. In R. J. Mammone, editor, *Neural networks for speech and image processing*, chapter 10. Chapman and Hall, London, 1994.

[427] T. N. Petersen, C. Lundegaard, M. Nielsen, H. Bohr, J. Bohr, S. Brunak, G. P. Gippert, and O. Lund. Prediction of protein secondary structure at 80% accuracy. *Proteins*, 41:17–20, 2000.

[428] P. A. Pevzner. *Computational Molecular Biology—An Algorithmic Approach.* MIT Press, Cambridge, MA, 2000.

[429] G. Pollastri, P. Baldi, P. Fariselli, and R. Casadio. Improved prediction of the number of residue contacts in proteins by recurrent neural networks. *Bioinformatics*, 2001. Proceedings of the ISMB 2001 Conference.

[430] V. V. Prabhu. Symmetry observations in long nucleotide sequences. *Nucl. Acids Res.*, 21:2797–2800, 1993.

[431] J. W. Pratt, H. Raiffa, and R. Schlaifer. *Introduction to Statistical Decision Theory.* MIT Press, Cambridge, MA, 1995.

[432] S. R. Presnell and F. E. Cohen. Artificial neural networks for pattern recognition in biochemical sequences. *Ann. Rev. Biophys. Biomol. Struct.*, 22:283–298, 1993.

[433] S. J. Press. *Bayesian Statistics: Principles, Models, and Applications.* John Wiley, New York, 1989.

[434] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical recipes in C.* Cambridge University Press, Cambridge, 1988.

[435] L. G. Presta and G. D. Rose. Helix signals in proteins. *Science*, 240:1632-1641, 1988.

[436] W. C. Probst, L. A. Snyder, D. I. Schuster, J. Brosius, and S. C. Sealfon. Sequence alignment of the G-protein coupled receptor superfamily. *DNA and Cell Biol.*, 11:1-20, 1992.

[437] N. Qian and T. J. Sejnowski. Predicting the secondary structure of globular proteins using neural network models. *J. Mol. Biol.*, 202:865-884, 1988.

[438] M. B. Qumsiyeh. Evolution of number and morphology of mammalian chromosomes. *J. Hered.*, 85:455-465, 1994.

[439] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, 77:257-286, 1989.

[440] T. A. Rapoport. Transport of proteins across the endoplasmic reticulum membrane. *Science*, 258:931-936, 1992.

[441] M. G. Reese, D. Kulp, H. Tammana, and D. Haussler. Genie-gene finding in *drosophila melanogaster. Genome Res.*, 10:529-538, 2000.

[442] F. M. Richards and C. E. Kundrot. Identification of structural motifs from protein coordinate data: Secondary structure and first-level supersecondary structure. *Proteins*, 3:71-84, 1988.

[443] D. S. Riddle, J. V. Santiago, S. T. Bray-Hall, N. Doshi, V.P. Grantcharova, Q. Yi, and D. Baker. Functional rapidly folding proteins from simplified alphabets. *Nat. Struct. Biol.*, 4:805-809, 1997.

[444] R. Riek, S. Hornemann, G. Wider, M. Billeter, R. Glockshuber, and K. Wuthrich. NMR structure of the mouse prion protein domain PrP(121-321). *Nature*, 382:180-182, 1996.

[445] S. K. Riis and A. Krogh. Improving prediction of protein secondary structure using structured neural networks and multiple sequence alignments. *J. Comput. Biol.*, 3:163-183, 1996.

[446] J. J. Rissanen. Modeling by shortest data description. *Automatica*, 14:465-471, 1978.

[447] É. Rivals, M. Dauchet, J. P. Delahaye, and O. Delgrange. Compression and genetic sequence analysis. *Biochimie*, 78:315-322, 1996.

[448] D. Ron, Y. Singer, and N. Tishby. The power of amnesia. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 6, pages 176-183. Morgan Kaufmann, San Francisco, CA, 1994.

[449] G. D. Rose. Protein folding and the Paracelsus challenge. *Nat. Struct. Biol.*, 4:512-514, 1997.

[450] G. D. Rose and T. P. Creamer. Protein folding: predicting predicting. *Proteins*, 19:1-3, 1994.

[451] B. Rost and C. Sander. Improved prediction of protein secondary structure by use of sequence profiles and neural networks. *Proc. Nat. Acad. Sci. USA*, 90:7558-7562, 1993.

[452] B. Rost and C. Sander. Prediction of protein secondary structure at better than 70% accuracy. *J. Mol. Biol.*, 232:584-599, 1993.

[453] B. Rost and C. Sander. Combining evolutionary information and neural networks to predict protein secondary structure. *Proteins*, 19:55-72, 1994.

[454] B. Rost, C. Sander, and R. Schneider. Redefining the goals of protein secondary structure prediction. *J. Mol. Biol.*, 235:13-26, 1994.

[455] D. E. Rumelhart, R. Durbin, R. Golden, and Y. Chauvin. Backpropagation: The basic theory. In *Backpropagation: Theory, Architectures and Applications*, pages 1-34. Lawrence Erlbaum Associates, Hillsdale, NJ, 1995.

[456] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, and the PDP Research Group, editors, *Parallel distributed processing: Explorations in the microstructure of cognition.*, volume 1: Foundations, pages 318-362, Cambridge, MA., 1986. MIT Press.

[457] R. Russell and G. Barton. The limits of protein secondary structure prediction accuracy from multiple sequence alignment. *J. Mol. Biol.*, 234:951-957, 1993.

[458] J. R. Saffran, R. N. Aslin, and E. L. Newport. Statistical learning by 8-month-old infants. *Science*, 274:1926-1928, 1996.

[459] Y. Sakakibara. Efficient learning of context-free grammars from positive structural examples. *Info. Comput.*, 97:23-60, 1992.

[460] Y. Sakakibara, M. Brown, R. Hughey, I. S. Mian, K. Sjölander, R. C. Underwood, and D. Haussler. Stochastic context-free grammars for tRNA modeling. *Nucl. Acids Res.*, 22:5112-5120, 1994.

[461] S. L. Salzberg, A. L. Delcher, S. Kasif, and O. White. Microbial gene identification using interpolated Markov models. *Nucl. Acids Res.*, 26:544-548, 1998.

[462] C. Sander and R. Schneider. Database of homology-derived protein structures and the structural meaning of sequence alignment. *Proteins*, 9:56-68, 1991.

[463] F. Sanger, G. M. Air, B. G. Barrell, N. L. Brown, A. R. Coulson, C. A. Fiddes, C. A. Hutchison, P. M. Slocombe, and M. Smith. Nucleotide sequence of bacteriophage phi X174 DNA. *Nature*, 265:687-695, 1977.

[464] D. Sankoff and P. Rousseau. Locating the vertices of a Steiner tree in an arbitrary metric space. *Math. Prog.*, 9:240-246, 1975.

[465] L. K. Saul and M. I. Jordan. Exploiting tractable substructures in intractable networks. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 486-492. MIT Press, Cambridge, MA, 1996.

[466] M. A. Savageau. Power-law formalism: a canonical nonlinear approach to modeling and analysis. In V. Lakshmikantham, editor, *World Congress of Nonlinear Analysts 92*, volume 4, pages 3323-3334. Walter de Gruyter Publishers, Berlin, 1996.

[467] R. D. Schachter. Probabilistic inference and influence diagrams. *Operation Res.*, 36:589-604, 1988.

[468] R. D. Schachter, S. K. Anderson, and P. Szolovits. Global conditioning for prob-

abilistic inference in belief networks. In *Proceedings of the Uncertainty in AI Conference*, pages 514–522, San Francisco, CA, 1994. Morgan Kaufmann.

[469] D. Schneider, C. Tuerk, and L. Gold. Selection of high affinity RNA ligands to the bacteriophage r17 coat protein. *J. Mol. Biol.*, 228:862–869, 1992.

[470] F. Schneider. Die funktion des arginins in den enzymen. *Naturwissenschaften*, 65:376–381, 1978.

[471] R. Schneider, A. de Daruvar, and C. Sander. The HSSP database of protein structure-sequence alignments. *Nucleic Acids Res.*, 25:226–230, 1997.

[472] T. D. Schneider. Reading of DNA sequence logos: Prediction of major groove binding by information theory. *Meth. Enzymol.*, 274:445–455, 1996.

[473] T. D. Schneider and R. M. Stephens. Sequence logos: A new way to display consensus sequences. *Nucl. Acids Res.*, 18:6097–6100, 1990.

[474] T. D. Schneider, G. D. Stormo, L. Gold, and A. Ehrenfeucht. Information content of binding sites on nucleotide sequences. *J. Mol. Biol.*, 188:415–431, 1986.

[475] B. Scholkopf, C. Burges, and V. Vapnik. Extracting support data for a given task. In U. M. Fayyad and R. Uthurusamy, editors, *Proceedings First International Conference on Knowledge Discovery and Data Mining*. AAAI Press, Menlo Park, CA, 1995.

[476] H. P. Schwefel and R. Manner, editors. *Parallel Problem Solving from Nature*, Berlin, 1991. Springer-Verlag.

[477] R. R. Schweitzer. Anastasia and Anna Anderson. *Nat. Genet.*, 9:345, 1995.

[478] W. Schwemmler. *Reconstruction of Cell Evolution: A Periodic System of Cells*. CRC Press, Boca Raton, FL, 1994.

[479] D. B. Searls. Linguistics approaches to biological sequences. *CABIOS*, 13:333–344, 1997.

[480] T. J. Sejnowski and C. R. Rosenberg. Parallel networks that learn to pronounce English text. *Complex Syst.*, 1:145–168, 1987.

[481] P. H. Sellers. On the theory and computation of evolutionary distances. *SIAM J. Appl. Math.*, 26:787–793, 1974.

[482] H. S. Seung, H. Sompolinsky, and N. Tishby. Statistical mechanics of learning from examples. *Phys. Rev. A*, 45:6056–6091, 1992.

[483] C. E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27:379–423, 623–656, 1948.

[484] R. Sharan and R. Shamir. CLICK: a clustering algorithm with applications to gene expression analysis. In *Proceedings of the 2000 Conference on Intelligent Systems for Molecular Biology (ISMB00), La Jolla, CA*, pages 307–316. AAAI Press, Menlo Park, CA, 2000.

[485] I. N. Shindyalov, N. A. Kolchanov, and C. Sander. Can three-dimensional contacts in protein structures be predicted by analysis of correlated mutations? *Prot. Eng.*, 7:349–358, 1994.

[486] J. E. Shore and R. W. Johnson. Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy. *IEEE Trans. Info. Theory*, 26:26–37, 1980.

[487] P. Sibbald and P. Argos. Weighting aligned protein or nucleic acid sequences to correct for unequal representation. *J. Mol. Biol.*, 216:813-818, 1990.

[488] R. R. Sinden. *DNA Structure and Function.* Academic Press, San Diego, 1994.

[489] K. Sjölander, K. Karplus, M. Brown, R. Hughey, A. Krogh, I. S. Mian, and D. Haussler. Dirichlet mixtures: a method for improved detection of weak but significant protein sequence homology. *CABIOS*, 12:327-345, 1996.

[490] A. F. Smith and G. O. Roberts. Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods. *J. R. Statis. Soc.*, 55:3-23, 1993.

[491] A. F. M. Smith. Bayesian computational methods. *Phil. Trans. R. Soc. London A*, 337:369-386, 1991.

[492] T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147:195-197, 1981.

[493] P. Smyth, D. Heckerman, and M. I. Jordan. Probabilistic independence networks for hidden Markov probability models. *Neural Comp.*, 9:227-267, 1997.

[494] E. E. Snyder and G. D. Stormo. Identification of protein coding regions in genomic DNA. *J. Mol. Biol.*, 248:1-18, 1995.

[495] V. V. Solovyev, A. A. Salamov, and C. B. Lawrence. Predicting internal exons by oligonucleotide composition and discriminant analysis of spliceable open reading frames. *Nucl. Acids Res.*, 22:5156-5153, 1994.

[496] V. V. Solovyev, A. A. Salamov, and C. B. Lawrence. Prediction of human gene structure using linear discriminant functions and dynamic programming. In C. Rawling, D. Clark, R. Altman, L. Hunter, T. Lengauer, and S. Wodak, editors, *Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology*, pages 367-375, Cambridge, 1995. AAAI Press.

[497] E. L. L. Sonnhammer, S. R. Eddy, and R. Durbin. Pfam: a comprehensive database of protein domain families based on seed alignments. *Proteins*, 28:405-420, 1997.

[498] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher. Comprehensive identification of cell cycle-regulated genes of the yeast *saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell*, 9:3273-3297, 1998.

[499] D. J. Spiegelhalter, A. P. Dawid, S. L. Lauritzen, and R. G. Cowell. Bayesian analysis in expert systems. *Stat. Sci.*, 8:219-283, 1993.

[500] F. Spitzer. Markov random fields and Gibbs ensembles. *Am. Math. Monthly*, 78:142-154, 1971.

[501] S. Stamm, M. Q. Zhang, T. G. Marr, and D. M. Helfman. A sequence compilation and comparison of exons that are alternatively spliced in neurons. *Nucl. Acids Res.*, 22:1515-1526, 1994.

[502] S. Steinberg, A. Misch, and M. Sprinzl. Compilation of tRNA sequences and sequences of tRNA genes. *Nucl. Acids Res.*, 21:3011-3015, 1993.

[503] G. Stoesser, P. Sterk, M. A. Tull, P. J. Stoehr, and G. N. Cameron. The EMBL nucleotide sequence database. *Nucl. Acids Res.*, 25:7-13, 1997.

[504] A. Stolcke and S. Omohundro. Hidden Markov model induction by Bayesian model merging. In S. J. Hanson, J. D. Cowan, and C. Lee Giles, editors, *Advances*

in *Neural Information Processing Systems*, volume 5, pages 11-18. Morgan Kaufmann, San Mateo, CA, 1993.

[505] P. Stolorz, A. Lapedes, and Y. Xia. Predicting protein secondary structure using neural net and statistical methods. *J. Mol. Biol.*, 225:363-377, 1992.

[506] G. D. Stormo, T. D. Schneider, L. Gold, and A. Ehrenfeucht. Use of the "perceptron" algorithm to distinguish translational initiation sites in *e. coli. Nucl. Acids Res.*, 10:2997-3011, 1982.

[507] G. D. Stormo, T. D. Schneider, and L. M. Gold. Characterization of translational initiation sites in *e. coli. Nucl. Acids Res.*, 10:2971-2996, 1982.

[508] C. D. Strader, T. M. Fong, M. R. Tota, and D. Underwood. Structure and function of G protein-coupled receptors. *Ann. Rev. Biochem.*, 63:101-132, 1994.

[509] R. Swanson. A unifying concept for the amino acid code. *Bull. Math. Biol.*, 46:187-203, 1984.

[510] R. H. Swendsen and J. S. Wang. Nonuniversal critical dynamics in Monte Carlo simulations. *Phys. Rev. Lett.*, 58:86-88, 1987.

[511] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E. S. Lander, and T. R. Golub. Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. *Proc. Natl. Acad. Sci. USA*, 96:2907-2912, 1999.

[512] R. E. Tarjan and M. Yannakakis. Simple linear-time algorithms to test the chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs. *SIAM J. Computing*, 13:566-579, 1984.

[513] R. L. Tatusov and E. V. Koonin D. J. Lipman. A genomic perspective on protein families. *Science*, 278:631-637, 1997.

[514] F. J. R. Taylor and D. Coates. The code within the codons. *Biosystems*, 22:177-187, 1989.

[515] W. R. Taylor and K. Hatrick. Compensating changes in protein multiple sequence alignments. *Prot. Eng.*, 7:341-348, 1994.

[516] T. A. Thanaraj. A clean data set of EST-confirmed splice sites from Homo sapiens and standards for clean-up procedures. *Nucl. Acids Res.*, 27:2627-2637, 1999.

[517] H. H. Thodberg. A review of Bayesian neural networks with an application to near infrared spectroscopy. *IEEE Trans. Neural Networks*, 7:56-72, 1996.

[518] C. A. Thomas. The genetic organization of chromosomes. *Ann. Rev. Genet.*, 5:237-256, 1971.

[519] J. L. Thorne, H. Kishino, and J. Felsenstein. An evolutionary model for maximum likelihood alignment of DNA sequences. *J. Mol. Evol.*, 33:114-124, 1991.

[520] L. Tierney. Markov chains for exploring posterior distributions. *Ann. Statis.*, 22:1701-1762, 1994.

[521] I. Tinoco, Jr., O. C. Uhlenbeck, and M. D. Levine. Estimation of secondary structure in ribonucleic acids. *Nature*, 230:362-367, 1971.

[522] D. M. Titterington, A. F. M. Smith, and U. E. Makov. *Statistical Analysis of Finite Mixture Distributions*. John Wiley & Sons, New York, 1985.

[523] N. Tolstrup, C. V. Sensen, R. A. Garrett, and I. G. Clausen. Two different

and highly organized mechanisms of translation initiation in the archaeon sulfolobus solfataricus. *Extremophiles*, 4:175-179, 2000.

[524] N. Tolstrup, J. Toftgard, J. Engelbrecht, and S. Brunak. Neural network model of the genetic code is strongly correlated to the GES scale of amino-acid transfer free-energies. *J. Mol. Biol.*, 243:816-820, 1994.

[525] E. N. Trifonov. Translation framing code and frame-monitoring mechanism as suggested by the analysis of mRNA and 16S rRNA nucleotide sequences. *J. Mol. Biol.*, 194:643-652, 1987.

[526] M. K. Trower, S. M. Orton, I. J. Purvis, P. Sanseau, J. Riley, C. Christodoulou, D. Burt, C. G. See, G. Elgar, R. Sherrington, E. I. Rogaev, P. St George-Hyslop, S. Brenner, and C. W. Dykes. Conservation of synteny between the genome of the pufferfish (*Fugu rubripes*) and the region on human chromosome 14 (14q24.3) associated with familial Alzheimer disease (AD3 locus). *Proc. Natl. Acad. Sci. USA*, 93:1366-1369, 1996.

[527] D. H. Turner and N. Sugimoto. RNA structure prediction. *Ann. Rev. Biophys. Biophys. Chem.*, 17:167-192, 1988.

[528] E. C. Uberbacher and R. J. Mural. Locating protein-coding regions in human DNA sequences by a multiple sensor-neural network approach. *Proc. Natl. Acad. Sci. USA*, 88:11261-11265, 1991.

[529] E. C. Uberbacher, Ying Xu, and R. J. Mural. Discovering and understanding genes in human DNA sequence using GRAIL. *Meth. Enzymol.*, 266:259-281, 1996.

[530] J. van Helden, B. Andre, and J. Collado-Vides. Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *J. Mol. Biol.*, 281:827-842, 1998.

[531] J. van Helden, M. del Olmo, and J. E. Perez-Ortin. Statistical analysis of yeast genomic downstream sequences reveals putative polyadenylation signals. *Nucl. Acids Res.*, 28:1000-1010, 2000.

[532] E. P. van Someren, L. F. A. Wessels, and M. J. T. Reinders. Linear modeling of genetic networks from experimental data. In *Proceedings of the 2000 Conference on Intelligent Systems for Molecular Biology (ISMB00), La Jolla, CA*, pages 355-366. AAAI Press, Menlo Park, CA, 2000.

[533] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.

[534] B. Venkatesh, B. H. Tay, G. Elgar, and S. Brenner. Isolation, characterization and evolution of nine pufferfish (*Fugu rubripes*) actin genes. *J. Mol. Biol.*, 259:655-665, 1996.

[535] J. Vilo and A. Brazma. Mining for putative regulatory elements in the yeast genome using gene expression data. In *Proceedings of the 2000 Conference on Intelligent Systems for Molecular Biology (ISMB00), La Jolla, CA*, pages 384-394. AAAI Press, Menlo Park, CA, 2000.

[536] M. Vingron and P. Argos. A fast and sensitive multiple sequence alignment algorithm. *CABIOS*, 5:115-121, 1989.

[537] E. O. Voit. *Canonical Nonlinear Modeling*. Van Nostrand and Reinhold, New York, 1991.

[538] M. V. Volkenstein. The genetic coding of protein structure. *Biochim. Biophys. Acta*, 119:418–420, 1966.

[539] G. von Heijne. A new method for predicting signal sequence cleavage sites. *Nucl. Acids Res.*, 14:4683–4690, 1986.

[540] G. von Heijne. *Sequence Analysis in Molecular Biology: Treasure Trove or Trivial Pursuit?* Academic Press, London, 1987.

[541] G. von Heijne. Transcending the impenetrable: How proteins come to terms with membranes. *Biochim. Biophys. Acta*, 947:307–333, 1988.

[542] G. von Heijne. The signal peptide. *J. Membrane Biol.*, 115:195–201, 1990.

[543] G. von Heijne and C. Blomberg. The beta structure: Inter-strand correlations. *J. Mol. Biol.*, 117:821–824, 1977.

[544] P. H. von Hippel. Molecular databases of the specificity of interaction of transcriptional proteins with genome DNA. In R.F. Goldberger, editor, *Gene expression. Biological regulation and Development, vol. 1*, pages 279–347, New York, 1979. Plenum Press.

[545] S. S. Wachtel and T. R. Tiersch. Variations in genome mass. *Comp. Biochem. Physiol. B*, 104:207–213, 1993.

[546] G. Wahba. *Spline Models of Observational Oata.* Society for Industrial and Applied Mathematics, Philadelphia, PA, 1990.

[547] J. Wang and R. H. Swendsen. Cluster Monte Carlo algorithms. *Physica A*, 167:565–579, 1990.

[548] J. Wang and W. Wang. A computational approach to simplifying the protein folding alphabet. *Nat. Struct. Biol.*, 6:1033–1038, 1999.

[549] Z. X. Wang. Assessing the accuracy of protein secondary structure. *Nat. Struct. Biol.*, 1:145–146, 1994.

[550] M. S. Waterman. *Introduction to Computational Biology.* Chapman and Hall, London, 1995.

[551] T. A. Welch. A technique for high performance data compression. *IEEE Computer*, 17:8–19, 1984.

[552] J. Wess. G-protein-coupled receptors: molecular mechanisms involved in receptor activation and selectivity of g-protein recognition. *FASEB J.*, 11:346–354, 1997.

[553] J. V. White, C. M. Stultz, and T. F. Smith. Protein classification by stochastich modeling and optimal filtering of amino-acid sequences. *Mathem. Biosci.*, 119:35–75, 1994.

[554] K. P. White, S. A. Rifkin, P. Hurban, and D. S. Hogness. Microarray analysis of *drosophila* development during metamorphosis. *Science*, 286:2179–2184, 1999.

[555] S. H. White. Global statistics of protein sequences: Implications for the origin, evolution, and prediction of structure. *Ann. Rev. Biophys. Biomol. Struct.*, 23:407–439, 1994.

[556] S. H. White and R. E. Jacobs. The evolution of proteins from random amino acid sequences. I. Evidence from the lengthwise distribution of amino acids in

modern protein sequences. *J. Mol. Evol.*, 36:79–95, 1993.

[557] J. Whittaker. *Graphical Models in Applied Multivariate Statistics.* John Wiley & Sons, New York, 1990.

[558] B. L. Wiens. When log-normal and gamma models give different results: a case study. *The American Statistician*, 53:89–93, 1999.

[559] K. L. Williams, A. A. Gooley, and N. H. Packer. Proteome: Not just a made-up name. *Today's Life Sciences*, June:16–21, 1996.

[560] E. Wingender, X. Chen, R. Hehl, H. Karas, I. liebich, V. Matys, T. Meinhardt, M. Pruss, I. Reuter, and F. Schacherer. TRANSFAC: an integrated system for gene expression regulation. *Nucl. Acids Res.*, 28:316–319, 2000.

[561] H. Winkler. *Verbreitung und Ursache der Parthenogenesis im Pflanzen und Tierreich.* Fischer, Jena, 1920.

[562] C. R. Woese. *The Genetic Code. The Molecular Basis for Genetic Expression.* Harper & Row, New York, 1967.

[563] C. R. Woese, D. H. Dugre, S. A. Dugre, M. Kondo, and W. C. Saxinger. On the fundamental nature and evolution of the genetic code. *Cold Spring Harbor Symp. Quant. Biol.*, 31:723–736, 1966.

[564] C. R. Woese and G. E. Fox. Phylogenetic structure of the prokaryotic domain: The primary kingdoms. *Proc. Natl. Acad. Sci. USA*, 74:5088–5090, 1977.

[565] C. R. Woese, R. R. Gutell, R. Gupta, and H. F. Noller. Detailed analysis of the higher-order structure of 16S-like ribosomal ribonucleic acids. *Microbiol. Rev.*, 47:621–669, 1983.

[566] R. V. Wolfenden, P. M. Cullis, and C. C. F. Southgate. Water, protein folding, and the genetic code. *Science*, 206:575–577, 1979.

[567] T. G. Wolfsberg, A. E. Gabrielian, M. J. Campbell, R. J. Cho, J. L. Spouge, and D. Landsman. Candidate regulatory sequence elements for cell cycle-dependent transcription in *saccharomyces cerevisiae. Genome Res.*, 9:775–792, 1999.

[568] D. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.

[569] J. T. Wong. A co-evolution theory of the genetic code. *Proc. Natl. Acad. Sci. USA*, 72:1909–1912, 1975.

[570] F. S. Wouters, M. Markman, P. de Graaf, H. Hauser, H. F. Tabak, K. W. Wirtz, and A. F. Moorman. The immunohistochemical localization of the non-specific lipid transfer protein (sterol carrier protein-2) in rat small intestine enterocytes. *Biochim. Biophys. Acta*, 1259:192–196, 1995.

[571] C. H. Wu. Artificial neural networks for molecular sequence analysis. *Comp. Chem.*, 21:237–256, 1997.

[572] C. H. Wu and J.W. McLarty. *Neural Networks and Genome Informatics.* Elsevier, Amsterdam, 2000.

[573] J. R. Wyatt, J. D. Puglisi, and I Tinoco, Jr. Hybrid system for protein secondary structure prediction. *BioEssays*, 11:100–106, 1989.

[574] L. Xu. A unified learning scheme: Bayesian-Kullback Ying-Yang machine. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8. MIT Press, Cambridge, MA, 1996.

[575] M. Ycas. The protein text. In H. P. Yockey, editor, *Symposium on information*

theory in biology, pages 70-102, New York, 1958. Pergamon.

[576] T. Yi, Y. Huang, M. I. Simon, and J. Doyle. Robust perfect adaptation in bacterial chemotaxis through integral feedback control. *Proc. Natl. Acad. Sci. USA*, 97:4649-4653, 2000.

[577] H. P. Yockey. *Information Theory and Molecular Biology*. Cambridge University Press, Cambridge, 1992.

[578] J. York. Use of the Gibbs sampler in expert systems. *Artif. Intell.*, 56:115-130, 1992.

[579] C. H. Yuh, H. Bolouri, and E. H. Davidson. Genomic cis-regulatory logic: experimental and computational analysis of a sea urchin gene. *Science*, 279:1896-1902, 1998.

[580] A. Zemla, C. Venclovas, K. Fidelis, and B. Rost. A modified definition of SOV, a segment-based measure for protein secondary structure prediction assessment. *Proteins*, 34:220-223, 1999.

[581] M. Q. Zhang. Large-scale gene expression data analysis: a new challenge to computational biologists. *Genome Res.*, 9:681-688, 1999.

[582] X. Zhang, J. Mesirov, and D. Waltz. Hybrid system for protein secondary structure prediction. *J. Mol. Biol.*, 225:1049-1063, 1992.

[583] J. Zhu, J. Liu, and C. Lawrence. Bayesian adaptive alignment and inference. In T. Gaasterland, P. Karp, K. Karplus, C. Ouzounis, C. Sander, and A. Valencia, editors, *Proceedings of Fifth International Conference on Intelligent Systems for Molecular Biology*, pages 358-368. AAAI Press, 1997. Menlo Park, CA.

[584] A. Zien, R. Kuffner, R. Zimmer, and T. Lengauer. Analysis of gene expression data with pathway scores. In *Proceedings of the 2000 Conference on Intelligent Systems for Molecular Biology (ISMB00), La Jolla, CA*, pages 407-417. AAAI Press, Menlo Park, CA, 2000.

[585] M. Zuker. Computer prediction of RNA structure. *Meth. Enzymol.*, 180:262-288, 1989.

[586] M. Zuker and P. Stiegler. Optimal computer folding of large RNA sequences using thermodynamic and auxiliary information. *Nucl. Acids Res.*, 9:133-148, 1981.

[587] M. Zvelebil, G. Barton, W. Taylor, and M. Sternberg. Prediction of protein secondary structure and active sites using the alignment of homologous sequences. *J. Mol. Biol.*, 195:957-961, 1987.

基本词汇英汉对照表

accession number	识别号码
human	个人识别号码
active sampling	主动采样
aging	衰老
Alice in Wonderland	艾丽斯漫游仙境
alpha-helix	α螺旋
alphabet	字符集，符号表
merged	合并字符集
reduced	简化字符集
alternative splicing	可变剪接
Altschul, S. F.	S · F · 阿特休尔
amino acids	氨基酸
codons	氨基酸密码子
composition	氨基酸组成
dihedral angle	氨基酸二面角
encoding	氨基酸编码
genetic code	氨基酸遗传密码
GES scale	氨基酸GES标度
glycosylation	氨基酸糖基化
hydrophobicity	氨基酸疏水性
in beta-sheets	β折叠中的氨基酸

in helix	螺旋结构中的氨基酸
in HMM	隐马氏模型中的氨基酸
orthogonal encoding	氨基酸正交编码
pathways	氨基酸合成途径
substitution matrices	氨基酸替换矩阵
Anastasia	阿纳斯塔西娅
ancestor	祖先
Anderson, A.	A·安德森
antique DNA（aDNA）	古DNA
Arabidopsis thaliana	阿布属拟南芥
archaon	原始的
asymmetric window	不对称窗口
background information	背景知识
backpropagation	反向传播
adaptive	自适应反向传播
learning order	反向传播的学习顺序
bacteria	细菌
bacteriophage	噬菌体
bacteriorhodopsin	细菌视紫红质
Bayes theorem	贝叶斯定理
Bayesian framework	贝叶斯体系
belief	置信度
Bellman principle	贝尔曼原则
bendability	可弯曲性
beta breakers	β断点
beta-sheet	β折叠
blind prediction	盲预测
Blobel, G.	G·布洛贝尔
Bochner's theorem	Bochner定理
Boltzmann-Gibbs distribution	波耳兹曼-吉布斯分布
Boolean	布尔
algebra	布尔代数

functions	布尔函数
networks	布尔网络
brain	脑
content-addressable retrieval	内容寻址检索
memory	记忆容量
branch length	枝长
branch point	分支点
Burset, M.	M・博塞特
C-terminal	C端
C-value	C值
paradox	C值悖论
cancer	癌症
capping	加帽作用
Carroll, L.	L・卡罗尔
CASP	蛋白质结构预测技术评判
Cavalier-Smith, T.	T・卡维利亚-史密斯
Chapman-Kolmogorov relation	Chapman-Kolmogorov方程
Chargaff's parity rules	夏加夫奇偶校验准则
chimpanzee	黑猩猩
Chomsky hierarchy	乔姆斯基层次
Chomsky normal form	乔姆斯基范式
chromatin	染色质
chromosome	染色体
components	染色体组成
unstable	不稳定染色体
classification	分类
classification error	分类误差
Claverie, J-M.	J-M・克拉弗里
clustering	聚类
Cocke-Kasami-Younger-algorithm	CKY算法
codon	密码子
start	起始密码子

deduction	演绎
DEFINE program	DEFINE程序
development	发育
dice	骰子
digital data	数字数据
dinucleotides	二核苷酸
Dirichlet distribution	Dirichlet分布
discriminant function	判别函数
distribution	分布
Boltzmann-Gibbs	波耳兹曼-吉布斯分布
DNA	
arrays	DNA阵列
bending	DNA弯曲
binding sites	DNA结合位点
chip	DNA芯片
helix types	DNA螺旋类型
library	DNA文库
melting	DNA解链
melting point	DNA解链温度
periodicity	DNA周期性
reading frame	DNA阅读框周期性
symmetries	DNA对称性
DNA renaturation experiments	DNA复性实验
DNA sequencing	DNA测序
DSSP program	DSSP程序
dynamic programming	动态规划
multidimensional	多维动态规划
E. coli	大肠杆菌
e-mail	电子邮件
encoding	编码
adaptive	自适应编码

ensemble	集合
ensembles	模型集
entropy	熵
maximum	最大熵
relative	相对熵
ethics	伦理学
evidence	显著性、证据、信念
evolution	进化
genetic code	遗传编码进化
protein families	蛋白家族进化
evolutionary information	进化信息
evolutionary algorithms	进化算法
evolutionary events	进化事件
evolutionary relationships	进化关系
exon assembly	外显子组装
exon shuffling	外显子倒位
exon-exon junction	外显子接合处
exons	外显子
extreme value distribution	极值分布
feature table	特征表
Fisher kernels	Fisher核
FORESST	蛋白质家族的二级结构数据库
forward–backward procedure	前向—后向过程
free energy	自由能
functional features	功能特征
fungi	真菌
Gamow, G.	G・盖莫
GenBank	GenBank数据库
gene	基因
coregulated	共调控基因

number in organism	生物体中的基因数量
protein coding	编码蛋白质的基因
gene pool	基因池
GeneMark	GeneMark基因发现程序
GeneParser	GeneParser程序
genetic code	遗传密码
Genie	Genie程序
genome	基因组
circular	环状基因组
diploid	二倍体基因组
double stranded	双链基因组
haploid	单倍体基因组
human	人类基因组
mammalian	哺乳动物基因组
single stranded	单链基因组
size	基因组规模
GenomeScan	GenomeScan软件
GenScan	GenScan软件
Gibbs sampling	吉布斯采样
glycosylation	糖基化
GRAIL	GRAIL软件
Guigo, R.	R·吉哥
halting problem	停机问题
Hansen, J.	J·汉森
hidden variables	隐变量
Hinton, G. E.	G·E·欣顿
histone	组蛋白
HMM	隐马氏模型
used in word and language modeling	用于文字和语言建模的隐马氏模型
Hobohm algorithm	Hobohm算法
homology	同源性

homology building	同源模建
HSSP	HSSP数据库
Hugo, V.	V·雨果
human	人类
human genome	人类基因组
chromosome size	人类基因组染色体大小
size	人类基因组规模
hybrid models	混合模型
hybridization	杂交
hydrogen bond	氢键
hydrophobicity	疏水性
signal peptide	信号肽疏水性
hydrophobicity scale	疏水性标度
hyperparameters	超参数
hyperplane	超平面
hypothesis	假设
complex	复杂假设
immune system	免疫系统
induction	归纳
infants	婴儿
inference	推断
input representation	输人表示
inside–outside algorithm	内部—外部算法
inteins	内含肽
intron	内含子
splice sites	内含子剪接位点
inverse models	逆模型
Jacobs, R. E.	R·E·雅各布斯
Johannsen, W.	W·约翰森
Jones, D.	D·琼斯
k-means algorithm	k均值聚类法

Kabsch, W.	W · 卡布希
Kernel methods	核方法
knowledge-based network	基于知识的网络
Krogh, A.	A · 克罗
Lagrange multiplier	拉格朗日算子
language	语言
computer	计算机语言
natural	自然语言
spelling	语言拼写
learning	学习
supervised	有监督学习
unsupervised	无监督学习
learning rate	学习率
likelihood	似然（似然度）
likelihood function	似然函数
linguistics	语言学
lipid environment	脂环境
lipid membrane	脂膜
liposome-like vesicles	类脂质体
loss function	损失函数
machine learning	机器学习
mammoth	猛犸象
map	地图
MAP estimate	最大后验估计
MaxEnt	最大熵
membrane proteins	膜蛋白
MEME	MEME程序
Mercer's theorem	Mercer定理
metabolic networks	代谢网络
Metropolis algorithm	Metropolis算法
generalizations	Metropolis算法推广
microarray expression data	微阵列表达数据

PAM matrix	PAM矩阵
parameters	参数
emission	生成参数
transition	转移参数
parse tree	分析树
partition function	分割函数
pathway	通路
PDB	PDB数据库
perceptron	感知器
multilayer	多层感知器
Petersen, T. N.	T·N·彼得森
Pfam	Pfam数据库
phase transition	相变
phonemes	音素
phosphorylation	磷酸化
phylogenetic information	系统进化信息
phylogenetic tree	系统进化树
plants	植物
polyadenylation	聚腺苷酸化
polymorphism	多态性
position-specific scoring matrices	位置特异性分值矩阵
posttranslational modification	翻译后修饰
prior	先验概率，先验分布
conjugate	共轭先验分布
Dirichlet	Dirichlet先验分布
gamma	伽玛先验分布
Gaussian	高斯先验分布
use in hybrid architectures	用于混合体系的高斯先验分布
uniform	均匀先验分布
profile	序列谱，分布图
bending potential	可弯曲性分布图
emission	生成谱

promoter	启动子
propositions	命题
PROSITE	PROSITE数据库
protein	蛋白质
beta-sheet	蛋白质β折叠
beta-sheet partners	蛋白质β折叠伴侣
helix	蛋白质螺旋
helix periodicity	蛋白质螺旋周期性
length	蛋白质长度
networks	蛋白质网络
secondary structure	蛋白质二级结构
secretory	分泌性蛋白
tertiary structure	蛋白质三级结构
Protein Data Bank	PDB数据库
protein folding	蛋白质折叠
proteome	蛋白质组
pruning	剪枝法
Prusiner, S. B.	S·B·普鲁赛纳
pseudo-genes	伪基因
pseudoknots	伪结
PSI-BLAST	PSI-BLAST程序
PSI-PRED	PSI-PRED程序
Qian, N.	N·钱
quantum chemistry	量子化学
reading frame	阅读框
open	开放阅读框
reductionism	还原论
redundancy reduction	冗余约简
regression	回归
regularizer	正则因子
regulatory circuits	调控网络

relative entropy	相对熵
renaturation kinetics	复性动力学
repeats	重复序列
representation	表示
orthogonal	正交表示
semiotic	符号表示
ribosome	核糖体
ribosome binding sites	核糖体结合位点
Riis, S.	S·里斯
ROC curve	ROC曲线
Rost, B.	B·罗斯特
rules	规则
Chou-Fasman	Chou-Fasman规则
S. solfataricus	极端嗜热古细菌
Sander, C.	C·桑德
Schneider, R.	R·施奈德
Schneider, T.	T·施奈德
secretory pathway	分泌通路
Sejnowski, T. J.	T·J·塞诺斯基
sensitivity	敏感度
sequence	序列
data	序列数据
families	序列家族
logo	序列标识
sequence space	序列空间
Shine-Dalgarno sequence	夏因—达尔加诺序列
signal anchor	信号锚
signal peptide	信号肽
signalling networks	信号转导网络
SignalP	SignalP预测程序
simulated annealing	模拟退火算法
single nucleotide polymorphism	单核苷酸多态性

Smith–Waterman algorithm	Smith–Waterman算法
social security numbers	社会保险号
sparse encoding	稀疏编码
specificity	特异性
speech recognition	语音识别
splice site	剪接位点
splines	样条函数
SSpro	SSpro预测服务工具
statistical mechanics	统计力学
statistical model fitting	统计模型拟合
stochastic	随机
grammars	随机文法
sampling	随机抽样
units	随机神经元
Stormo, G.	G·斯托蒙
STRIDE program	STRIDE程序
string	字符串
Student distribution	学生氏分布
support vector machines	支持向量机
SWISS-PROT	SWISS-PROT数据库
systemic properties	系统特性
TATA-box	TATA框
threshold gate	阈值门
time series	时间序列
TMHMM	TMHMM方法
training	训练
balanced	平衡训练
transcription initiation	转录起始位点
transfer free energy	转移自由能
transfer function	激活函数
sigmoidal	sigmoidal激活函数
translation initiation	翻译起始位点

trinucleotides	三核苷酸
tsar, Nicholas Ⅱ	沙皇尼古拉二世
t-test	*t*检验
Turing machine	图灵机
halting problem	停机问题
twilight zone	模糊区域
validation	检验
VC dimension	VC维数
virus	病毒
visual inspection	视觉检查
Viterbi algorithm	Viterbi算法
von Heijne, G.	G · 冯海因
Watson, J. D.	J · D · 沃森
Watson–Crick basepair	沃森—克里克碱基对
weight	权重
decay	权重衰减
logo	权重标识
matrix	权重矩阵
sharing	权重共享
weighting scheme	权重赋值方法
White, S. H.	S · H · 怀特
winner-take-all	胜者通吃
Ycas, M.	M · 伊卡斯
yeast	酵母